INSIGHT

This Issue's Feature:

Unique Abilities of the Systems Engineer

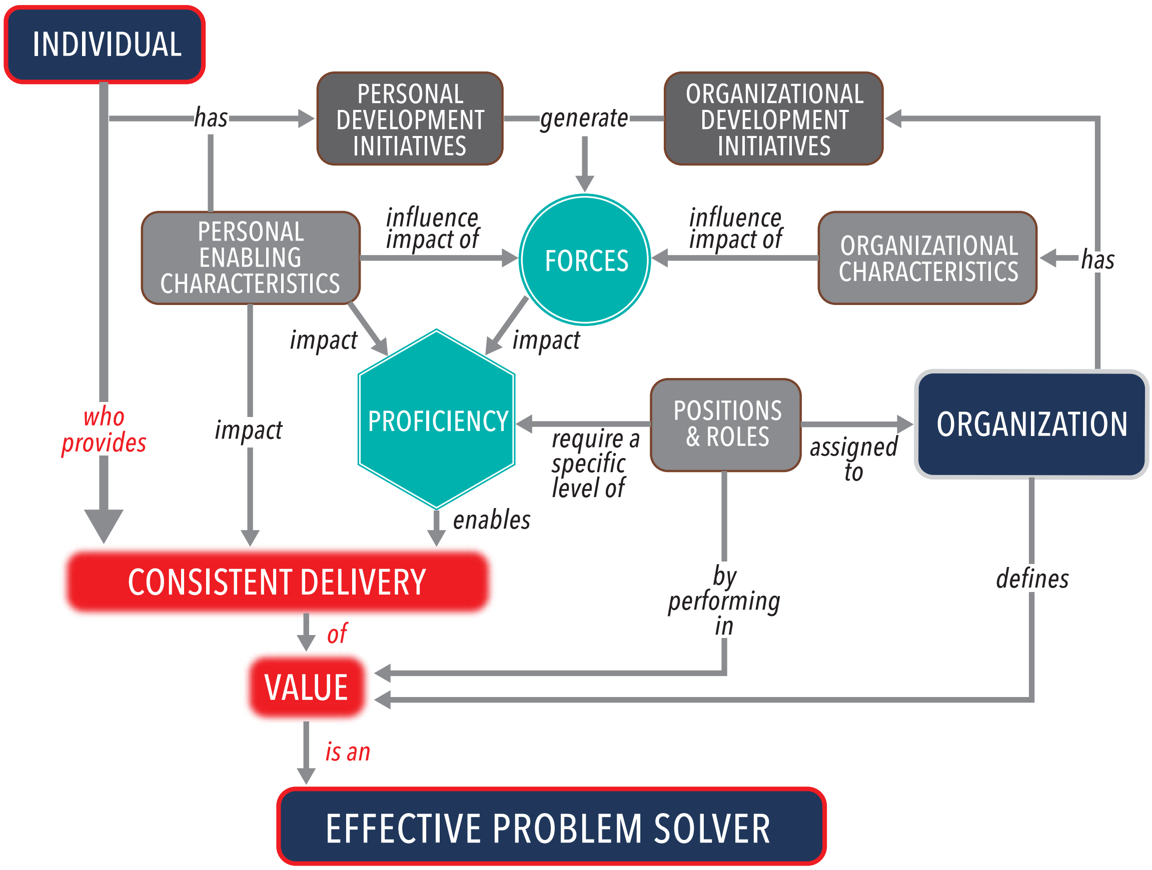


Illustration credit: from the article *Is Systems Engineering Effectiveness the Heart of Today's Employability Skills?* by Nicole Hutchison and Tom McDermott (page 14)

SEPTEMBER 2022
VOLUME 25 / ISSUE 3

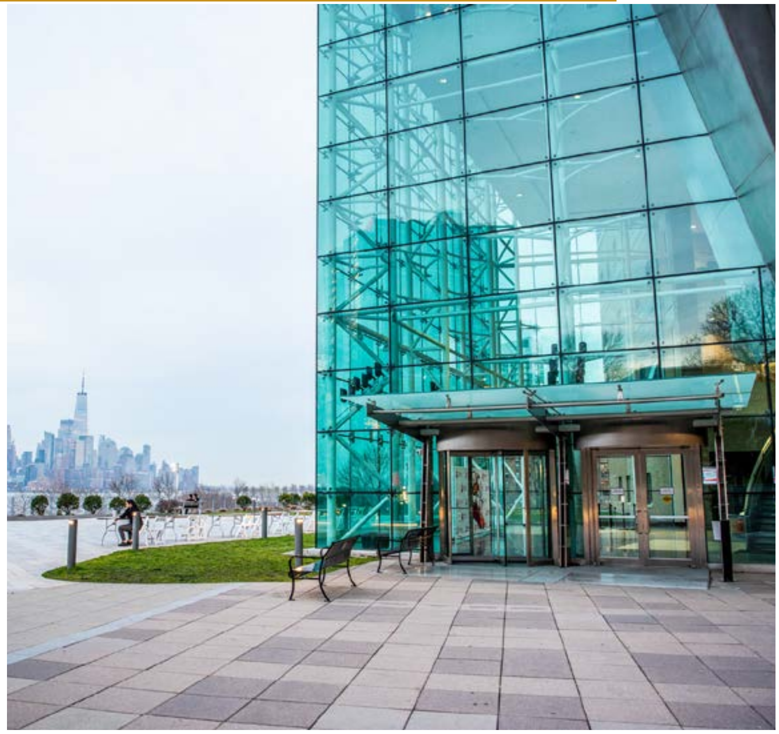
INCOSE

A PUBLICATION OF THE INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING

CAREER OPPORTUNITIES

STEVENS
INSTITUTE OF TECHNOLOGY
1870

School of Systems and Enterprises



The **School of Systems and Enterprises (SSE)** at Stevens Institute of Technology is seeking a pool of qualified adjuncts for a range of part-time teaching assignments in the areas of software engineering, systems analytics, industrial and systems engineering and engineering management, with openings beginning in **Summer 2022**.

Successful candidates will contribute to a dynamic and growing school that provides students with a research-centered interdisciplinary and transdisciplinary education embedded in systems thinking and design.

Visit **stevens.edu/careers** to apply.

INSIGHT

A PUBLICATION OF THE INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING

SEPTEMBER 2022 VOLUME 25 / ISSUE 3

Inside this issue

About This Publication

INFORMATION ABOUT INCOSE

INCOSE's membership extends to over 19,000 individual members and more than 200 corporations, government entities, and academic institutions. Its mission is to share, promote, and advance the best of systems engineering from across the globe for the benefit of humanity and the planet. INCOSE charters chapters worldwide, includes a corporate advisory board, and is led by elected officers and directors.

For more information, click here:
The International Council on Systems Engineering
(www.incose.org)

OVERVIEW

INSIGHT is the magazine of the International Council on Systems Engineering. It is published four times per year and features informative articles dedicated to advancing the state of practice in systems engineering and to close the gap with the state of the art. ***INSIGHT*** delivers practical information on current hot topics, implementations, and best practices, written in applications-driven style. There is an emphasis on practical applications, tutorials, guides, and case studies that result in successful outcomes. Explicitly identified opinion pieces, book reviews, and technology roadmapping complement articles to stimulate advancing the state of practice. ***INSIGHT*** is dedicated to advancing the INCOSE objectives of impactful products and accelerating the transformation of systems engineering to a model-based discipline.
Topics to be covered include resilient systems, model-based systems engineering, commercial-driven transformational systems engineering, natural systems, agile security, systems of systems, and cyber-physical systems across disciplines and domains of interest to the constituent groups in the systems engineering community: industry, government, and academia. Advances in practice often come from lateral connections of information dissemination across disciplines and domains. ***INSIGHT*** will track advances in the state of the art with follow-up, practically written articles to more rapidly disseminate knowledge to stimulate practice throughout the community.

Editor-In-Chief	William Miller
insight@incose.net	+1 908-759-7110
Assistant Editor	Lisa Hoverman
lisa.hoverman@incose.net	
Theme Editors	
Tom McDermott	tmcdermo@stevens.edu
Nicole Hutchison	nicole.hutchison@stevens.edu
Layout and Design	Chuck Eng
chuck.eng@comcast.net	
Member Services	INCOSE Administrative Office
info@incose.net	+1 858 541-1725

PERMISSIONS

Permissions Department – UK
John Wiley & Sons Ltd.
The Atrium,
Southern Gate,
Chichester
West Sussex, PO19 8SQ
UK
Email: Permissions@wiley.com
Fax: 44 (0) 1243 770620
or

Permissions Department – US
John Wiley & Sons Inc.
111 River Street MS 4-02
Hoboken, NJ 07030-5774
USA
Email: Permissions@wiley.com
Fax: (201) 748-6008

ARTICLE SUBMISSION

insight@incose.net

Publication Schedule. ***INSIGHT*** is published four times per year. Issue and article submission deadlines are as follows:
- June 2022 issue – 1 April 20022
- September 2022 issue – 1 July 2022
- December 2022 issue – 1 October 2022
- March 2023 issue – 2 January 2023

For further information on submissions and issue themes, visit the INCOSE website: www.incose.org

ISSN 2156-485X; (print) ISSN 2156-4868 (online)

ADVERTISE

Readership
INSIGHT reaches over 18,000 individual members and uncounted employees and students of more than 100 CAB organizations worldwide. Readership includes engineers, manufacturers/purchasers, scientists, research & development processionals, presidents and CEOs, students and other professionals in systems engineering.

Issuance	Circulation
2022, Vol 25, 4 Issues	100% Paid

Contact us for Advertising and Corporate Sales Services
We have a complete range of advertising and publishing solutions professionally managed within our global team. From traditional print-based solutions to cutting-edge online technology the Wiley-Blackwell corporate sales service is your connection to minds that matter. For an overview of all our services please browse our site which is located under the Resources section. Contact our corporate sales team today to discuss the range of services available:

- Print advertising for non-US journals
- Email Table of Contents Sponsorship
- Reprints
- Supplement and sponsorship opportunities
- Books
- Custom Projects
- Online advertising

Click on the option below to email your enquiry to your nearest office:

- Asia & Australia corporatesalesaustralia@wiley.com
- Europe, Middle East & Africa (EMEA) corporatesaleseurope@wiley.com
- Japan corporatesalesjapan@wiley.com
- Korea corporatesaleskorea@wiley.com

USA (also Canada, and South/Central America):

- Healthcare Advertising corporatesalesusa@wiley.com
- Science Advertising Ads_sciences@wiley.com
- Reprints Commercialreprints@wiley.com
- Supplements, Sponsorship, Books and Custom Projects busdev@wiley.com

Or please contact:
Marcom@incose.net

CONTACT

Questions or comments concerning:

Submissions, Editorial Policy, or Publication Management
Please contact: William Miller, Editor-in-Chief
insight@incose.org

Advertising—*please contact:*
Marcom@incose.net

Member Services – ***please contact:*** info@incose.org

ADVERTISER INDEX September Volume 25-3

CORPORATE ADVISORY BOARD — MEMBER COMPANIES

FROM THE EDITOR-IN-CHIEF

William Miller, insight@incose.net

We are pleased to announce the September 2022 *INSIGHT* issue published cooperatively with John Wiley & Sons as the systems engineering practitioners' magazine. The *INSIGHT* mission is to provide informative articles on advancing the practice of systems engineering and to close the gap between practice and the state of the art as advanced by *Systems Engineering*, the Journal of INCOSE also published by Wiley.

The issue theme is the *unique abilities of the systems engineer*. We thank theme editors Tom McDermott and Nicole Hutchison, both with the Systems Engineering Research Center (SERC) operated by Stevens Institute of Technology for the US Department of Defense, and the diversity of authors for their contributions. Tom and Nicole abstract the unique abilities of the systems engineer in their theme editorial and provide a brief synopsis for each of the themed articles that follow. The articles discuss the unique abilities of the systems engineer, and how they inform a world demanding core skills like leadership, systems thinking, innovation, and design. These articles address the generalization and application of systems engineering knowledge, skills, and competencies to challenges both inside and outside of our discipline. We hope this serves to inform the world to look toward systems engineering as a source to drive their future workforce strategies.

The September *INSIGHT* publishes a cautionary open letter to fellow systems engineers by INCOSE Fellow Michael Pennotti titled "Blinded by the Light?" Mike continues to be haunted by the two fatal aircraft crashes that took the lives of 346 passengers and crew in October 2018 and March 2019. Mike rhetorically asks how is it that systems engineers armed with our formal processes, digital engineering, agile methods, sophisticated models and even our own modeling language could have missed the design fault in the aircraft?

Your editor recalls accident reports of engineers accepting *normal risks* in US space shuttle missions that led to the loss of the Challenger and Columbia crews, tolerated because nothing catastrophically bad occurred in previous spaceflight missions. The Challenger and Columbia accident investigation reports faulted the organizational structures and interfaces that in effect allowed the two catastrophes to occur. There were in effect internal and external *air gaps* between management and engineering across NASA and its contractors, with no integrated, systemic view of safety, which was fragmented in a reductionist manner.

The Systems Engineering Vision 2035 published by INCOSE in early 2022, freely accessible at https://www.incose.org/about-systems-engineering/se-vision-2035, cites the need to address systems engineering challenges including the change needed to have systems engineering included on agendas for industry and government leadership.

We hope you find *INSIGHT*, the practitioners' magazine for systems engineers, informative and relevant. Feedback from readers is critical to *INSIGHT*'s quality. We encourage letters to the editor at insight@incose.net. Please include "letter to the editor" in the subject line. *INSIGHT* also continues to solicit special features, standalone articles, book reviews, and op-eds. For information about *INSIGHT*, including upcoming issues, see https://www.incose.org/products-and-publications/periodicals#INSIGHT. For information about sponsoring *INSIGHT*, please contact the INCOSE marketing and communications director at marcom@incose.net. ■

Systems Engineering: The Journal of The International Council on Systems Engineering

Call for Papers

Systems Engineering is intended to be a primary source of multidisciplinary information for the systems engineering and management of products and services, and processes of all types.

Systems Engineering is the archival journal of, and exists to serve the following objectives of, the International Council on Systems Engineering (INCOSE):

- To provide a focal point for dissemination of systems engineering knowledge
- To promote collaboration in systems engineering education and research
- To encourage and assure establishment of professional standards for integrity in the practice of systems engineering
- To improve the professional status of all those engaged in the practice of systems engineering
- To encourage governmental and industrial support for research and educational programs that will improve the systems engineering process and its practice

The journal will accept and review submissions in English from any author, in any global locality, whether or not the author is an INCOSE member. A body of international peers will review all submissions, and the reviewers will suggest potential revisions to the author.

Editorial selection of works for publication will be made based on content, without regard to the stature of the authors. Selections will include a wide variety of international works, recognizing and supporting the essential breadth and universality of the field. Final selection of papers for publication, and the form of publication, shall rest with the editor.

Submission of quality papers for review is strongly encouraged. The review process is estimated to take three months, occasionally longer for hard-copy manuscript.

Systems Engineering operates an online submission and peer review system that allows authors to submit articles online and track their progress, throughout the peer-review process, via a web interface. All papers submitted to *Systems Engineering*, including revisions or resubmissions of prior manuscripts, must be made through the online system. Contributions sent through regular mail on paper or emails with attachments will not be reviewed or acknowledged.

All manuscripts must be submitted online to *Systems Engineering* at ScholarOne Manuscripts, located at

http://mc.manuscriptcentral.com/SYS

Full instructions and support are available on the site, and a user ID and password can be obtained on the first visit.

Blinded by the Light?

An Open Letter to My Fellow Systems Engineers

Michael Pennotti, mcpennotti@icloud.com

Dear Colleague,

Despite the passage of time, I continue to be haunted by the Boeing 737 MAX. As I'm sure you are aware, the MAX was Boeing's newest jetliner when it was involved in two fatal crashes that took the lives of 346 passengers and crew in October 2018 and March 2019.

As a frequent flyer with more than a million air miles, I have vivid memories of several times when flying through severe turbulence caused a sudden drop in altitude that produced a collective gasp throughout the passenger cabin. I shudder to think of what the cabins of the two aircraft sounded like during the final minutes of their flights as the crews attempted to wrest control of their doomed aircraft back from an automated system intent on driving the nose of the planes into the sea (in the case of Lion Air) and the ground (in that of Ethiopian Airlines). I suspect we are fortunate that airplane black boxes do not include a cabin voice recorder.

As a system engineering practitioner and educator for more than 50 years, I am horrified that no member of our profession was able to prevent these tragedies, which I consider to be the worst system failures of our generation. And system failures they were — at many levels — as thoroughly documented in the FAA's "Return to Service" report of November 2018, Peter Robison's book *Flying Blind* and the Netflix documentary *Downfall*, as well as numerous press reports. In all the reporting, much is made of the fact that the flight control software used data from only one of the aircraft's two Angle of Attack (AoA) sensors. This introduced a single-point failure, something to avoid at all costs in a safety-critical system. Less widely noted is that in the two cases in question, data from the second sensor would not have been necessary to save the aircraft. The AoA data recovered from the airplanes' flight data recorders after the crashes proves this.

Comparing the AoA data from the two sensors in the Lion Air case indicates that the sensor driving the software had a constant bias of +20 degrees from the time the plane left the gate until the crash. Thought to be the result of a calibration error when a previously damaged sensor was replaced that means the software was being told that the plane was inclined upwards at 20 degrees during taxi and while on the runway, a situation that is not physically possible. This is not how airports are built! (It might be noted that 20 degrees exceeds the steepest slopes in the Tour de France.) Had the software been designed to recognize clearly erroneous inputs it would have recognized this input as such.. in that case, lacking any other information, the best the software could have done was to return control of the plane to the crew. Since the data from the second AoA sensor, unavailable during the flight but captured on the flight data recorder, indicates the plane's attitude was always nominal, the plane was never in jeopardy, and the crew should have been able to fly it without incident.

In the Ethiopian Airlines case, the two sensors remained in agreement for the first two minutes of the flight. At that point, the data from the sensor driving the software suddenly jumped from approximately +15 degrees to +75, then remained above +60 degrees until the crash. Once again, this was obviously an error. Neither the step function jump in attitude nor its reported value are physically possible. Once again, had this been recognized, the data from the second sensor indicate the crew should have been able to continue the flight without incident.

How is it that systems engineers armed with our formal processes, digital engineering, agile methods, sophisticated models and even our own modeling language could have missed this fault in the aircraft's design? What were we looking at? It reminds me of the story of the drunk found looking for his keys under a streetlamp rather than in the darkened park where he lost them because the light was better where he was. What are we missing and what do we have to learn from this case to ensure that such a tragedy never occurs again?

Michael Pennotti, PhD
INCOSE Fellow

Editorial of *INSIGHT* Special Feature

The Unique Abilities of the Systems Engineer

Tom McDermott, tmcdermo@stevens.edu; and **Nicole Hutchison,** nicole.hutchison@stevens.edu

The concept of "pi-shaped" skills and abilities are inherent to systems engineering. Most successful practitioners gain depth in at least one foundational discipline/domain and add a second area of depth in the discipline of systems engineering to go with breadth across technical, stakeholder, leadership, business, and other professional areas. The world is catching on, and with the rise of technology and automation many business leaders have started a call to "go pi -shaped" in both roles and training. In fact, workforce surveys across a variety of disciplines refer to the systems engineering core and professional skills recognized in the *INCOSE Competency Framework* (2018) as "employability skills." The discipline of systems engineering has emphasized the creation and value of these skills since its inception.

So, what can the world learn from systems engineering? Can we generalize the unique skills and abilities of the systems engineer across all of education and training — from early education to lifelong learning — to meet the needs of future workers?

In this issue, we present a set of articles that to discuss the unique abilities of the systems engineer, and how they inform a world demanding core skills like leadership, systems thinking, innovation, and design. These articles address the generalization and application of systems engineering knowledge, skills, and competencies to challenges both inside and outside of our discipline. We hope this serves to inform the world to look toward systems engineering as a source to drive their future workforce strategies.

In the first article, *Is Systems Engineering Effectiveness the Heart of Today's Employability Skills?* Tom McDermott and Nicole Hutchison (the editors of this special issue) discuss their work extending the results of the Helix study to a general framework for training and assessment of employability skills. Helix is a multi-year research study targeted at identifying the critical proficiencies of effective systems engineers. We found there is a significant correlation between Helix results and prevalent employability skills frameworks. Employability skills are the general skills that are necessary for success in the labor market at all employment levels and in all sectors and easily transfer from job to job. Other names for them are "soft," "workforce readiness," "career readiness," or "21st-century" skills. Learning and assessment of systems engineering skills can inform a parallel model for learning and assessment of employability skills. This article sets the theme for this special issue: how the unique abilities of the systems engineer can inform any organization's workforce development strategy, particularly when the organization tackles complex problems.

Clifford Whitcomb, Corina White, and Rabia Khan present *"Using the INCOSE Systems Engineering Competency Framework."* This article introduces the INCOSE Systems Engineering Competency Framework (SECF) and INCOSE Systems Engineering Assessment Guide (SECAG) and provides examples for using these in practice. The authors present four usage scenarios that demonstrate how organizations use the framework and assessment guide in practice for any company or enterprise looking to gain value from their workforce organization.

Caitlyn Singam next discusses INCOSE's SySTEAM initiative in "*A vision for universal & standardized access to systems competency education.*" The author notes that systems thinking and systems engineering competencies are not generally recognized by the academic community as being integral portions of science, technology, engineering, arts, and mathematics (STEAM) curricula, especially in pre-collegiate education, even though these disciplines are all unified by a common ability to understand and tackle problems from a holistic, integrated (systems-level) perspective. The article introduces the INCOSE SySTEAM Initiative (incose.org/system), an international volunteer-led effort, which is working on a vision of systems competencies in every STEAM classroom. This article discusses existing issues relating to systems competencies in education, as well as the efforts of the INCOSE SySTEAM Initiative to address those issues and facilitate change.

In "*The Digital Engineering Competency Framework (DECF): Leaning into Digital Transformation to Work in the 21st Century*," issue editor Nicole Hutchison and co-author Hoong Yan See Tao discuss the development of the Digital Engineering Competency Framework (DECF), the context of digital transformation and the trends for increasingly digital and digitized approaches to systems engineering. The DECF contains many of the skillsets that will become critical for anyone working in a digital engineering or digital acquisition space.

Fred Y. Robinson discusses the subjective nature of systems engineering achievement evaluation and certification, and a perspective on enhancements to the current programs for the development of systems engineering professionals in, "*For the Journey to Expertise in SE, Enhance the Path with Shu Ha Ri*." Shu Ha Ri, established in ancient practices such as martial arts, represents an approach for considering three phases of mastery development in systems engineering. Shu Ha Ri represents three phases of mastery pursuit, rooted in a triplet of ancient Japanese constructs simplified as Shu to learn the basics, Ha to learn tools, and Ri to extend beyond just using the tools and demonstrating skills. The author notes that the Western norm to chunk knowledge or skills into discrete levels or stages can constrain one's potential for flexibility or agility of continual learning, while the Eastern view of gaining knowledge in a particular Shu, Ha, or Ri phase promotes learning basics in new areas concurrently as we hone skills or evolve new techniques in others. Systems engineers, as expert practitioners who address wicked problems, have characteristics that both build upon and depart from their foundational learning, rather than the deliberate honing of deeper learning found in disciplines more focused on kind problems. Systems engineering, and Shu Ha Ri, may provide a basis for workforce development in all organizations and disciplines that deal with complex or wicked problems.

Richard Beasley discusses systems engineering roles and competencies across all disciplines in "*What's the role of a Systems Engineer in an Engineering Organisation?*" In this article, Beasley contends that systems engineering must be a long-term objective to make systems engineering the way we do engineering. Systems engineering is an integrating discipline, pulling together the more "traditionally" specialist engineering roles. In systems, engineering disciplines are interdependent and successful systems rely on the success of all disciplines. In the author's organization, they do this by embedding systems engineering processes in all disciplines and training all engineers in systems engineering and systems thinking.

In "*Systems Skills: From Here to Diversity*," Alan Harding explores how the abilities of systems engineers should contribute to diversity, equity, and inclusion. Systems engineers enable the development and sustainment of successful systems, working effectively across the disciplines, and engaging widely with stakeholders. Systems engineers are already diverse in outlook, but improvement is always possible. The author presents definitions of diversity, equity, and inclusion (DEI) and relates them to insights from various systems engineering related competency frameworks.

In "*Systems Engineers – Value Added Product Owners*," author Aswin Nair discusses why systems engineers make excellent product managers and product owners in agile product/service development and delivery. In agile development, product managers and product owners function as the bridge between the business and stakeholders and the product delivery teams. This requires specific skills such as having a customer-centric, design-thinking mindset and a good vision of the overall system, its capabilities, and the ability to enable cross-functional teams with the right information at the right time. Organizations and programs can improve overall quality and execution efficiency in their products and services by employing systems engineers as product managers and owners in today's "agile" world.

In "*Why Mountain Bike Trails Try to Scare You Off*," Courtney Wright provides a compelling argument as to why systems engineering certification has relevance to everything. This article goes far from the commonly discussed domains and products of systems engineering and looks at how a person might improve how they lead a bike ride by using systems engineering principles. It concludes with whether applying systems engineering in such an unusual case is enough to qualify a candidate to be an INCOSE Certified Systems Engineering Professional (CSEP).

The special issue concludes with issue editor Tom McDermott and co-author Molly Nadolski discussing their experiences "*Teaching Systems Engineering Practices Using Principles from Studio Art Education.*" The authors discuss a cross-disciplinary learning program that incorporates studio art, systems thinking, and systems architecture to improve systems competencies in any individual. The art studio format encourages the students to build their skills across a portfolio of work focused on the creation of appropriate views of a system, communicating them, and gaining critical review – three core aspects of the systems thinking process applicable to any discipline. Systems engineering concepts are easy to teach but more difficult to put into practice. The authors teach systems skills, methods, and tools in a studio art setting to create an environment for iterative practice.

As illustrated in INCOSE's *Vision 2035*, the discipline of systems engineering will continue to increase in relevance as complexity and interconnection increase. The novel skills that set systems engineers apart and enable us to deal with complexity and uncertainty are more relevant today than ever before – not only for us but for the wider world. ■

ABOUT THE AUTHORS

Tom McDermott serves as the Deputy Director and Chief Technology Officer of the Systems Engineering Research Center (SERC) at Stevens Institute of Technology in Hoboken, NJ. The SERC is a University Affiliated Research Center sponsored by the Office of the Secretary of Defense for Research and Engineering. With the SERC he develops new research strategies and is leading research on Digital Engineering transformation, education, security, and artificial intelligence applications. Mr. McDermott also teaches system architecture concepts, systems thinking and decision making, and engineering leadership. He consults with several organizations on enterprise modeling for transformational change and often serves as a systems engineering expert on government major program reviews. He currently serves on the INCOSE Board of Directors as Director of Strategic Integration.

Dr. Nicole Hutchison is a Principal Investigator (PI) and research engineer at the Systems Engineering Research Center (SERC). Her primary work through the SERC has been in human capital development research. This has included the development of competency frameworks for systems engineering (the Helix project), digital engineering, and mission engineering. Currently, Dr. Hutchison is the PI for the Simulation Training Environment for Digital Engineering (STEDE), a project that is developing realistic models to be used in training the DoD acquisition workforce in a way that builds hands-on digital engineering skills. She previously served on the BKCASE research team, which resulted in the development of the *Guide to the Systems Engineering Body of Knowledge* (SEBoK). She is currently the Managing Editor for the SEBoK and the Lead Editor for the "Enabling Systems Engineering" section of the SEBoK. Before joining the SERC, she spent 5 years working for Analytic Services, Inc., supporting the US Departments of Defense, Homeland Security, Health and Human Services, and Justice. She holds a PhD in systems engineering from Stevens and is a Certified Systems Engineering Professional (CSEP) through INCOSE.

Is Systems Engineering Effectiveness the Heart of Today's Employability Skills?

Nicole Hutchison, nicole.hutchison@stevens.edu; and **Tom McDermott,** tmcdermo@stevens.edu

ABSTRACT

There are three essential and interrelated sets of personal characteristics and skills that are acquired by the most valued and effective individuals as they develop their careers. These are (1) **self-leadership and learning**, (2) **complex problem solving**, and (3) **team leadership and collaboration**. These are interdependent and develop iteratively through practice in authentic work-related contexts. They are at the heart of the Helix study which was conducted by the Systems Engineering Research Center to discover the sets of proficiencies that make a systems engineer effective in their roles (Hutchison and Verma 2018). They are also at the heart of the "employability skills" so desired by today's businesses. Each of these three skillsets is supported by tools of different types and can be embedded into education and organizational learning programs. In this article we relate the development of effective systems engineers as a model for all professionals and frame a set of learning objectives that individuals and organizations can use to accelerate desired employability skills.

EMPLOYABILITY SKILLS

What are employability skills? Employability skills are defined as the general skills that are necessary for success in the labor market at all employment levels and in all sectors and easily transfer from job to job. They may be referred to as "soft", "workforce readiness", "career readiness", or "21st-century" skills. The Office of Career, Technical, and Adult Education (CTE) of the U.S. Department of Education recently published a framework for the education of employability skills. Figure 1 depicts the framework:

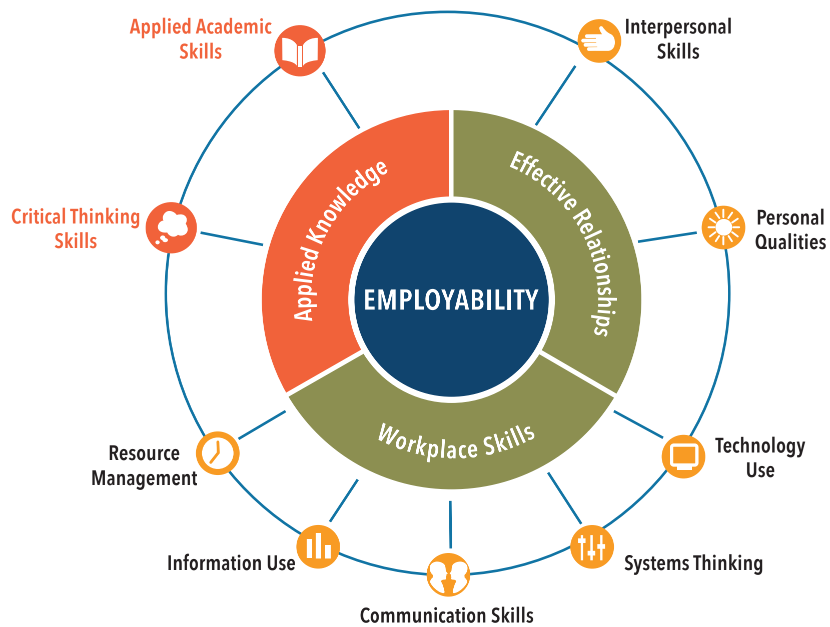


Figure 1. U.S. Department of Education framework for education of employability skills (https://cte.ed.gov/initiatives/employability-skills-framework)

Table 1 identifies the primary components of each of these skills from the CTE framework. The first three, systems thinking, critical thinking, and communication, are directly cited in the International Council on Systems Engineering (INCOSE) Competency Framework (PERKINS 2020). Each employability skill set in this table would also be recognized by most practic-

Table 1. Employability Skills and Primary Skill Components (https://cte.ed.gov/initiatives/employability-skills-framework)

Employability Skill	Skill Components
Systems Thinking	Understands systems and uses systems principles, collaborates in teams, assesses progress, evolves and adapts solutions
Critical Thinking	Thinks critically, thinks creatively, reasons about problems and solutions, makes sound decisions, solves problems, plans and organizes approaches to solve problems
Communication	Communicates verbally, listens actively and responds appropriately, observes carefully, comprehends written material, conveys information in writing and visually
Applied Academics	Effective applies reading, writing, math strategies, and scientific method to work strategies
Resource Management	Manages time, manages money, manages personnel, manages other resources
Information Use	Locates information, organizes information, uses information, analyzes information, communicates relevant information to others
Technology Use	Understands technology, uses different types of technology, recognizes how technology can be employed to solve problems
Interpersonal Skills	Understands teamwork and works with others, shows leadership, negotiates, respects individual differences, responds to others' needs
Personal Qualities	Accepts responsibility, exhibits self-discipline, takes initiative, works and learns independently, shows willingness to learn, adaptive and flexible, demonstrates professionalism, positive attitude, sense of self-worth, takes responsibility for professional growth

ing systems engineers as critical proficiencies in their roles.

In Table 1 the green text reflects a skill, and the purple text reflects a personal characteristic or disposition that allows them to use each skill effectively. Highly effective systems engineers develop all these skills in their careers. The systems engineering community can and should play a central role in bringing these skills to every professional workplace role, first by recognizing the relationships and then by supporting systems-related learning at all education levels.

THE HELIX STUDY

The Helix project is a multi-year research study targeted at identifying the proficiencies of effective systems engineers. Helix interviewed hundreds of successful systems engineers in various career stages to derive a model for these proficiencies and personal and organizational characteristics that enable them. Helix additionally worked with organizations to find the organizational characteristics that enable development and proficiency in systems engineering (Hutchison et al. 2018). The proficiencies are summarized in Figure 2.

Helix found these proficiencies were developed through a generative set of personal and organizational forces that reflect the characteristics of both individuals and the organization (Pyster, Hutchison, and Henry 2018). These characteristics are summarized in Figure 3.

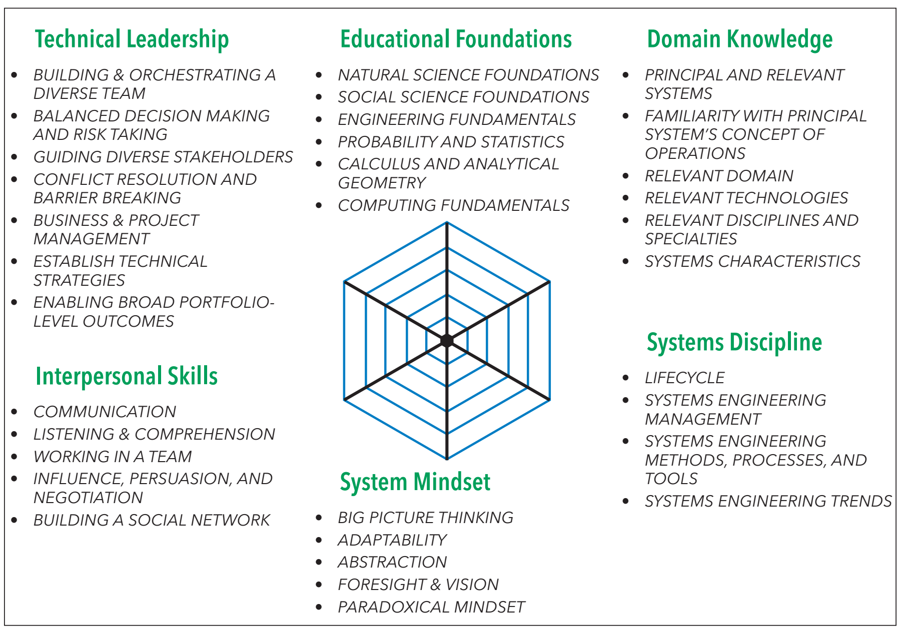


Figure 2. Helix Proficiency Areas

GENERALIZING THE HELIX FRAMEWORK TO EMPLOYABILITY SKILLS

There is a strong correlation between the systems engineering skills of Figures 2 and 3 and the employability skills of Figure 1 and Table 1. Systems engineering was created as a discipline for managing complex systems and projects across teams of people. Employers desire above all else people who can adapt and grow in a complex and adaptive world. Experienced systems engineers exhibit advanced complex problem-solving, self-leadership, and team collaboration skills. Employers also desire above all else these skills.

Recent research has identified that the Helix proficiency assessment toolset (helix-se.org), when generalized to non-systems engineering roles, provides a general framework for the development

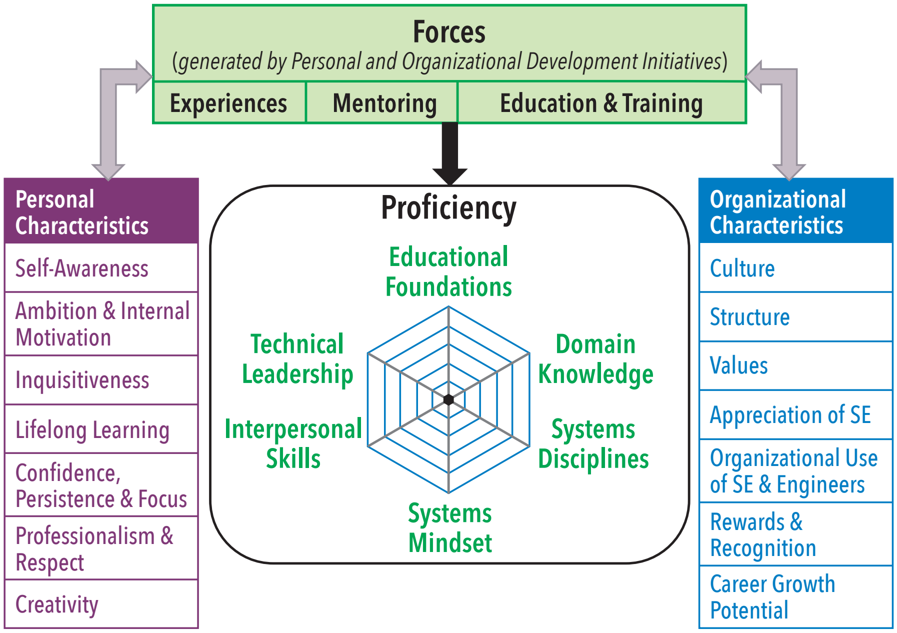


Figure 3. The Complete Helix Proficiency Framework

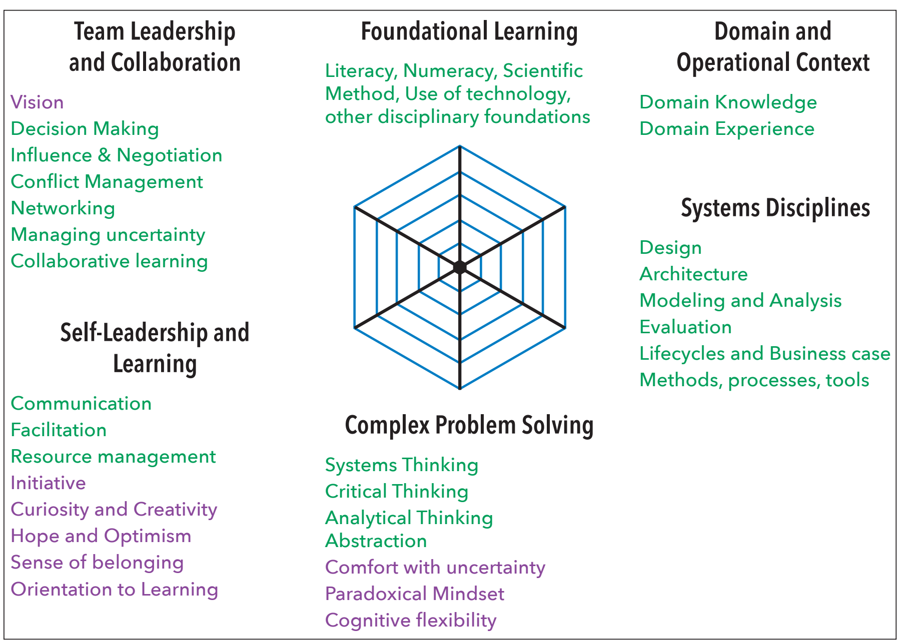


Figure 4. Helix framework extended to employability skills

and assessment of employability skills. The six Helix proficiency areas and personal characteristics of Figures 2 and 3 are at the heart of the CTE employability skills framework. The resulting combined framework is shown in Figure 4.

Employee development in any professional career is a progression of proficiencies starting from **foundational learning** to **experience in domain** to expansion into **systems responsibilities and disciplines**. Further development extends to the individual characteristics and skills developed with experience and learning on the job that support **complex problem solving, self-leadership and learning,** and **team leadership and collaboration**.

Again, in Figure 4 the green text reflects a skill or proficiency, and the purple text reflects a personal characteristic or disposition. The importance of the Helix framework is how it is applied in an organizational learning environment to create and assess the desired individual proficiencies. Each of the proficiencies listed in Figure 4 should be intentionally developed and assessed at the institutional level, in both education and business contexts.

WHY EMPLOYABILITY SKILLS ASSESSMENT IS IMPORTANT

The generalized Helix model correlates strongly with widely published data on employer desired skills. Figure 5 is the World Economic Forum's list of top-10 most desired employer work skills over the last 15 years, as organized by the employability skills proficiency areas (color codes).

The World Economic Forum (WEF) Future of Jobs reports survey global human resources and strategy experts a step to understanding shifting employer needs over time (WEF 2016, WEF 2020). Although the lists change over time, the themes remain clear.

Using the WEF list of 'desired skills for the future workforce' as a well-researched example, we can categorize the development of employability skills into four learning sets: an individual's foundational skills, an individual's relationship to **themselves** (self-leadership and personal learning), an individual's relationship to their **environment** (complex problem solving); and an individual's relationship to **others** (team leadership and collaboration).

Adaptive Learning/Learning Strategies and Complex Problem Solving are highlighted in Figure 5 as these are the core aspects Systems and Critical Thinking. Note that neither of these are narrowly focused skills. These are defined as:

- **Complex Problem Solving**: using a collection of processes and activities related to the cognitive, emotional, and motivational aspects of ourselves, applied to dynamic situations, to achieve ill-defined goals (Dorner).
- **Adaptive Learning**: transforming ourselves from being a passive receptor of information to being an active collaborator in our learning process (Crick).

In addition, the generalization of interpersonal skills and personal qualities in the employability skills framework can be further detailed as:

- **Self-Leadership and Personal Learning**: capabilities associated with a person's orientation to learning – how they respond to risk, uncertainty and challenge and their ability to purposefully 'learn their way forwards' to design, engage, fail, and learn and generate new knowledge which improves or transforms the job to be done.
- **Team Leadership and Collaboration**: capabilities associated with individual and group collaboration learning capacity as it is manifested in relationships between people who are aligned around achieving a shared purpose of value. This is about collaborating in

World Economic Forum Top 10 Skills Survey Results

Most In-Demand Skills of 2015	Most In-Demand Skills of 2020	Most In-Demand Skills of 2025
1. Complex Problem Solving	**1. Complex Problem Solving**	1. Analytical Thinking and Innovation
2. Coordination with Others	**2. Critical Thinking**	***2. Active Learning/Learning Strategies***
3. People Management	**3. Creativity**	***3. Complex Problem Solving***
4. Critical Thinking	4. People Management	**4. Critical Thinking and Analysis**
5. Negotiation	5. Coordination with Others	**5. Creativity, Originality and Initiative**
6. Quality Control	6. Emotional Intelligence	6. Leadership and Social Influence
7. Service Orientation	7. Judgment and Decision Making	7. Technology Use, Monitoring and Control
8. Judgment and Decision Making	8. Service Orientation	8. Technology Design and Prototyping
9. Active Listening	9. Negotiation	9. Resilience, Stress tolerance, Flexibility
10. Creativity	10. Cognitive Flexibility	10. Reasoning, Problem-solving, Ideation

Figure 5. World Economic Forum most In-Demand Skills Rankings

teams to identify problems, conceptualize broad responses, and compose successful new solutions which add value for the stakeholders. It is about the ability to conceptualize (model), plan for, and successfully implement transformative change.

The Helix study further determined a list of these value "products" which are relevant to the complex problem-solving domain (Pyster, Hutchison, and Henry 2018). These can be generalized as:

- maintaining team, organizational, and product purpose and vision,
- translating between technical and operational language,
- enabling team collaboration and learning,
- managing the "emergence" of the solution or response to a problem,
- enabling good and holistic decisions, and
- supporting the business case for the system.

A learning program design should effectively train these skills and products. Helix cited three personal enabling characteristics that impact the development of proficiencies in complex problem-solving. These are a paradoxical mindset, abstraction, and comfort with uncertainty.

- **Paradoxical mindset** – the ability to simultaneously entertain conflicting or contradictory thoughts – was identified as a core determining characteristic which separates good systems people from others in the Helix study (Pyster, Hutchison, and Henry 2018).
- **Abstraction** – identifying the right abstractions and levels and communicating meaningful abstractions to others – is a central skill to understanding problems in complex situations and communicating potential problem/solution sets.
- **Comfort with uncertainty** is an underlying psychological and physiological trait that can affect our ability to address problems in complexity. Each of these characteristics can be assessed in individuals so that they can learn about themselves and improve over time.

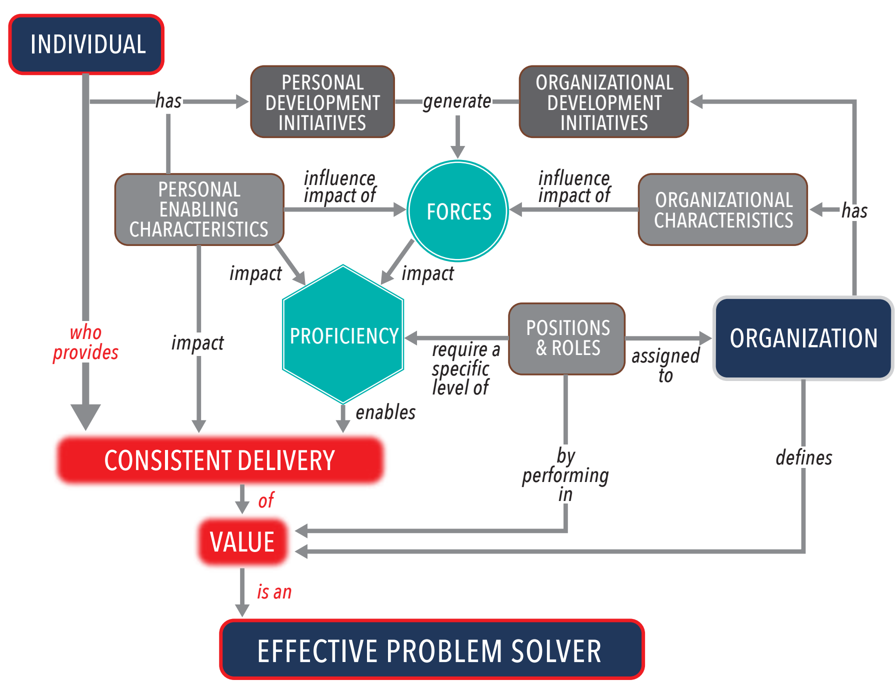


Figure 6. Organizational and personal "forces" build complex problem-solving skills

Table 2. Employability Skills and Primary Learning Concepts

Learning concept	Related activities	Employability skills
Sensemaking	Asking critical questions, collecting information and knowledge, seeing others' perspectives, visualizing problems, and solutions in context, learning by doing	Critical thinking, locates, organizes analyzes, and uses information, communicates relevant information to others, thinks critically, thinks creatively, systems thinking, reasons about problems and solutions, makes sound decisions, plans and organizes approaches to solve problems, understands systems and uses systems principles
Adaptive Thinking and Adaptive Learning	Initiative or motivation (agency) to learn, is curious, does it in a creative environment, accepts coaching and coaches others, develops sense of belonging to a group, remains optimistic (Crick)	Manages time and other resources, accepts responsibility, exhibits self-discipline, takes initiative, works and learns independently, shows willingness to learn, adaptive and flexible, demonstrates professionalism, positive attitude, sense of self-worth, takes responsibility for professional growth
Design Mindset	Approaches design as a human activity not a preconceived solution, form relationships with others, leads and follows, able to abstract and communicate ideas, asks good questions and collaborates in teams, assesses progress, iteratively evolves, and adapts solutions	Communicates verbally, listens actively and responds appropriately, observes carefully, comprehends written material, conveys information in writing and visually, understands teamwork and works with others, shows leadership, negotiates, respects individual differences, responds to others' needs,
Architectural Competence	Willingness to explore problems outside their experience, structures problems and solutions, learns and applies established patterns, defines the business/financial aspects of problem solving, interacts with stakeholders and translates that to project needs (at different organizational levels)	Links critical thinking to systems thinking to design to evaluation, structures information, employs math strategies, develops narratives, understands application of technology, uses different types of technology, recognizes how technology can be employed to solve problems
Analytical Approaches	Uses scientific method as a process to attack problems, effective applies reading, writing, modeling, and mathematical strategies to solving real problems	Scientific method, effective reading and writing skills, able to develop and use mathematical strategies, applies disciplinary methods and tools, has domain experience

HELIX IN ORGANIZATIONS

The organization influences the forces that build complex problem-solving proficiency through not just roles but enabling organizational characteristics and organizational development initiatives. These are shown in Figure 6. Helix discusses the generation of forces that affect the development of proficiencies.

A complex problem solver fills a position in their organization which is probably not titled or even focused on that aspect of their work. Attainment of roles often defines our focus at work. However, "systems thinker" and "complex problem solver" are not defined roles (and probably will never be). These activities are executed in many organizations across many domains, and the challenge is to help people develop the necessary skills broadly across an organization. An individual's personal development initiatives are often linked to their role, and systems engineering is a defined role in many organizations, but the employability skills inherent to systems engineers are often still not a focus of employee development. Specifically creating educational programs to advance "employability skills" and related may serve as an antidote to the primarily disciplinary learning that starts in educational institutions and tends to carry forward into business, as the concept of employability skills is easily understood by organizations and their human resource departments.

Another critical organizational force is an emphasis on the collaborative learning that comes from working in teams. The organization must support periods of reflection within team-oriented projects where the complexity of the situation can be reanalyzed, and solutions visualized. These serve as opportunities to view problems in complexity and for practicing adaptive learning. At the pace of today's business activities, it is easy for organizations to unintentionally create forces that inhibit the development of employability skills. Organizational leadership must establish and incentivize a collaborative environment; establish effective environments and times where individuals can share knowledge and create shared learning; support these data-driven activities that promote systems and critical thinking, use of information, use of technologies; and use these to promote and coach individual communication skills (McDermott 2019). In these team-based and outcome-driven activities, people can improve their personal qualities and interpersonal skills through more open-ended team activities. In summary, give a team of employees an open-ended problem related to the organization's domain, and use it as an opportunity to train and coach these types of employability skills (not just domain skills).

EDUCATION FOR EMPLOYABILITY SKILLS

Employability skills education must be focused on practical methods and tools that can be applied across many different roles. This is where systems thinking and systems engineering and the related "systems approach" can be applied generally across organizational development programs. Two activities in an appropriate framework can contribute significantly to the systems learning process: exploring a system architecture and developing multidisciplinary trades for decision making (McDermott and Freeman 2017). Table 2 lists one framework for organizing learning concepts in support of both systems and employability skills, using research on the way systems thinking influences systems engineering (book).

Organizations and educational institutions must create the environment to learn and use these employability skills. Helix makes use of the Quality of Interaction (Qi) Framework as an assessment tool for organizations, measuring cognitive diversity and psychological safety (Qi). Teams solve problems faster when they are more cognitively diverse, and team members are more generative and adaptive in an environment of psychological safety (Reynolds and Lewis 2018). Formal education and development of experience in the systems and critical thinking processes, methods, and tools are critical to learning or "generative" organizations (Senge 1990) and to organizational needs for greater adaptiveness, which might be described as architectural attributes. "Leader as architect" has long been described as core to organizational success (Collins 2001).

Applying employability skills while collaborating in teams is the force for skill development. Teams must be able to develop a mindset and processes to be creative and to visualize and communicate knowledge in complex situations. The organization must support periods of reflection within projects where the complexity of the situation can be reanalyzed, and solutions visualized (McDermott 2019). In the pace of today's business activities, it is easy for organizations to unintentionally create forces that inhibit the development of employability skills. Organizational leadership must establish and incentivize a collaborative environment, support data collection and analysis activities that support situational understanding of both internal and external context, establish effective environments and times where individuals can share knowledge and create shared learning, and promote creative narrative and storytelling along with analytical data (ICCPM 2012; McDermott 2019). What remains missing from most competency guidance is "create shared learning." Institutions must intentionally develop the problem-focused learning environments for employees to flourish.

APPLICATION AND CONCLUSION

Many employers desire workers with the self-leadership, collaboration, and complex problem-solving skills that are coming to be known as employability skills. Effective systems engineers develop both the technical skills in their disciplinary foundations and all the related skills and attributes highlighted in the CTE Employability Skills framework — they have well-developed employability skills. The Helix study found that effective systems engineers naturally develop employability skills over time and that these can be assessed in individuals and organizations. However, institutions must create the environment and structure the learning to focus on and accelerate development.

Imagine if the competencies and proficiencies of the systems engineer became the established model for all professionals and were trained at all levels from early education to workplace development. The INCOSE Competency Framework provides a basis for these skills and the Helix framework provides a competency-based assessment. What is needed is for the systems engineering community to bring these frameworks to the rest of the world — the unique abilities of the systems engineer need to be developed in everyone. ■

REFERENCES

- Collins, J. C. 2001. *Good to great: Why some companies make the leap ... and others don't*. New York, NY: HarperCollins Publishers.
- Perkins Collaborative Research Network. 2020. "Employability Skills." https://cte.ed.gov/initiatives/employability-skills-framework.
- Frank, M. 2006. "Knowledge, Abilities, Cognitive Characteristics and Behavioral Competences of Engineers with High Capacity for Engineering Systems Thinking (CEST)." *Systems Engineering* 9(2): 91-103.
- Godfrey, P., Deakin Crick, R., & Huang, S. 2014. "Systems Thinking, Systems Design and Learning Power in Engineering Education." *International Journal of Engineering Education* 30(1), pp. 112–127.
- SERC (Systems Engineering Research Center) Technical Report SERC-2018-TR-101-B. 2018. Atlas 1.1 Implementation Guide: Moving from Theory to Practice. Hoboken, US-NJ: Stevens Institute of Technology.
- Pyster, A., Hutchison, N., and Henry, D. 2018. *The Paradoxical Mindset of Systems Engineers: Uncommon Minds, Skills, and Careers*. New York, US-NY: Wiley.
- Reynolds, A. and Lewis, D. 2018. "The Two Traits of the Best Problem-Solving Teams." *Harvard Business Review*, 2 April. https://hbr.org/2018/04/the-two-traits-of-the-best-problem-solving-teams#comment-section.
- McDermott T. and Freeman D. 2016. "Systems Thinking in the Systems Engineering Process, New Methods and Tools." In *Systems Thinking: Foundation, Uses and Challenges*, edited by M. Frank, H. Shaked, and S. Koral-Kordova, Part 3. Hauppauge, US-NY: Nova Science Publishers.
- McDermott T. 2019. "Advanced Visualization Toolset." In *Evolving Toolbox for Complex Project Management*, edited by A. Gorod, L. Hallo, V. Ireland, and I. Gunawan, Chapter 17. Boca Raton, US-FL: Auerbach Publications.
- McDermott, T., Hutchison, N., and Crick R. 2021. "The Evolution of HELIX: A Competency Model for Complex Problem Solving." *INCOSE International Symposium* 31 (1): 907-925. https://doi.org/10.1002/j.2334-5837.2021.00877.x
- Senge, P. M. 1990. *The Fifth Discipline: The Art and Practice of the Learning Organization*.New York: Doubleday/Currency.
- WEF. 2016. *The Future of Jobs Report 2016:* Employment, Skills and Workforce Strategy for the Fourth Industrial Revolution. Geneva, Switzerland: World Economic Forum. January 2016. https://www3.weforum.org/docs/WEF_Future_of_Jobs.pdf
- WEF. 2020. *The Future of Jobs Report 2020*. Geneva, Switzerland: World Economic Forum. 20 October 2020. https://www3.weforum.org/docs/WEF_Future_of_Jobs_2020.pdf

> continued on page 29

Using the INCOSE Systems Engineering Competency Framework

Clifford Whitcomb, caw356@cornell.edu; **Corina White**, corina.white@nps.edu; and **Rabia Khan**, rhkhan@nps.edu

ABSTRACT
This article provides examples for using the INCOSE Systems Engineering Competency Framework in practice. Background information is summarized, competency model tailoring is explained, and four usage scenarios are presented. The usage scenarios demonstrate how employers use the framework and assessment guide. The four usage scenarios are job candidate recruitment, individual competency assessment, job description tailoring for an engineering job that needs systems engineering competencies, and job description tailoring for a systems engineering job requiring an update for model-based systems engineering and digital engineering competencies. All scenarios are based on those included in the Systems Engineering Competency Assessment Guide.

KEYWORDS: systems engineering competency framework, systems engineering competency assessment, systems engineering job description

INTRODUCTION

This article provides examples for using the INCOSE Systems Engineering Competency Framework (SECF) and the INCOSE Systems Engineering Assessment Guide (SECAG) in practice (Presland 2018 and Presland, Whitcomb, and Zipes 2022). The SECF documents an internationally available source of systems engineering competencies. The SECAG provides details of assessment criteria associated with each of the SECF competence areas. These documents are the framework and assessment guide, respectively. Background information is summarized, competency model tailoring is explained, and four usage scenarios are presented.

BACKGROUND

Competency is "a measure of an individual's ability in terms of their knowledge, skills, and behavior to perform a given role." (Holt and Perry 2011). Competency frameworks organize and structure competency definitions and relationships making them suitable for various purposes (IEEE P1484.20.2:2022). The INCOSE Competency Working Group (CWG) developed the framework "to define a global standard for those competencies regarded as central to the practice and profession of systems engineering, together with a set of indicators which can be used to verify attainment of those competencies" (Presland 2018). The importance of competency frameworks, the history of the framework and assessment guide development, and a top-level overview of the framework are summarized.

Importance of Competency Frameworks

Competency definitions and frameworks form the basis of crucial capabilities for organizations to establish and maintain an effective workforce. The development and usage of competency frameworks is important for the following reasons (IEEE P1484.20.2:2022):

- Promote a mutual understanding for those who use competencies for defining policies, standards, and data structures
- Promote reuse of competency definitions
- Provide well-defined competencies to inform specific functions for training, education, credentialing, organizations, employers, and government agencies
- Promote improved communications and alignment among participating stakeholders in the use of competencies
- Enable free movement of workers economically and geographically

Frameworks offer the opportunity to move to more holistic approaches for companies and enterprises to provide value from their workforce organization.

Framework Development

The Competency Working Group (CWG) developed the framework in 2011 immediately after chartering their group in 2010. The original aim of the CWG is to improve the practice of systems engineering through development of competency management approaches. The INCOSE UK SE Competency Framework/Guide (INCOSE UK 2010) and many other existing systems engineering competency models assisted

Table 1. INCOSE Systems Engineering Competency Framework

Core Competencies	Professional Competencies	Management Competencies	Technical Competencies
Core competencies underpin engineering as well as systems engineering.	Behavioral competencies well-established within the Human Resources (HR) domain.	The ability to perform tasks associated with controlling and managing Systems Engineering activities.	The ability to perform tasks associated primarily with the suite of Technical Processes identified in the INCOSE SE Handbook.
• Systems Thinking • Lifecycles • Capability Engineering • General Engineering • Critical Thinking • Systems Modeling and Analysis	• Communications • Ethics and Professionalism • Technical Leadership • Negotiation • Team Dynamics • Facilitation • Emotional Intelligence • Coaching and Mentoring	• Planning • Monitoring and Control • Decision Management • Concurrent Engineering • Business and Enterprise Integration • Acquisition and Supply • Information Management • Configuration Management • Risk and Opportunity Management	• Requirements Definition • System Architecting • Design for… • Integration • Interfaces • Verification • Validation • Transition • Utilization and Support • Retirement
Integrating Competencies			
This competency group recognizes Systems Engineering as an integrating discipline, joining activities, and thinking from specialists in other disciplines to create a coherent whole.			
• Project Management • Finance • Logistics • Quality			

CWG to develop the evolution of the current framework. The CWG collaborated extensively with the National Defense Industrial Association (NDIA), Systems Engineering Division, Education and Training Committee to develop a common approach to the definition of the framework. Six publications between 2014 and 2019 describe the details of the framework development (Gelosh et al. 2014, 2015, 2016, 2017, 2018, 2019). Three significant aspects were considered during early development, creating a role-based competency framework that is extensible, scalable, and tailorable by any customer organization, assigning competency categories, and forming a Professional category that covers leadership and "soft-skills" (Gelosh et al. 2015 and Beasley, Gelosh, and Pickard 2019). As development continued, CWG included specific additions of systems engineering competencies from U.S. defense systems acquisition competency models, systems planning, research, development, engineering (SPRDE), engineering (ENG), Better Buying Power 3.0, and the Systems Engineering Career Competency Model (SECCM) (Gelosh et al. 2016 and Whitcomb et al. 2017). The Systems Engineering Handbook Fourth Edition and the INCOSE Professional Certification programs both informed the framework development (Gelosh et al. 2016 and 2017). For a complete description of the competency sources for the development of the framework, see (Presland 2018). The end-result is a framework that is broadly usable for the global systems engineering community.

The assessment guide development began in 2019 after the publication of the framework. The assessment guide assists users in assessing systems engineering competencies characterized within the framework. The assessment guide provides detailed guidance to users of the framework to (Presland, Whitcomb, and Zipes 2022):

- Create a benchmark standard for each level of proficiency within each competence area
- Define a set of standardized terminology for competency indicators to promote like-for-like comparison
- Provide typical non-domain-specific indicators of evidence that users use to confirm experience in each competency area

The assessment guide provides a structure to extend the framework adding relevant sub-indicator classifications to each indicator of competence for knowledge, activity, and behaviors (as professional attitude), as appropriate. Each indicator of competence also includes examples of objective evidence of personal contribution to activities performed, or professional behaviors applied. The assessment guide provides guidance on how to evaluate individuals to establish their proficiency in the competencies defined in the framework and how to differentiate between proficiency at each of the five levels.

Systems Engineering Competency Framework

The framework provides a set of systems engineering competencies organized into five categories, shown Table 1.

Each competency has a title and a description, as well as a brief explanation of why that competency matters. Each competency has a list of effective indicators of knowledge and experience partitioned into five levels of increasing competence from awareness, supervised practitioner, practitioner, lead practitioner, to expert. The effective indicators are knowledge, skill, ability, and behavior entries. These entries

Table 2. Candidate Recruitment Desired Competencies for a System Architect Supervised Practitioner Position

Title	Description
Systems Thinking	The application of the fundamental concepts of systems thinking to Systems Engineering. These concepts include understanding what a system is, its context within its environment, its boundaries and interfaces, and that it has a lifecycle. System thinking applies to the definition, development, and production of systems within an enterprise and technological environment and is a framework for curiosity about any system of interest.
Ethics and Professionalism	Professional ethics encompass the personal, organizational, and corporate standards of behavior expected of systems engineers. Professional ethics also encompasses the use of specialist knowledge and skills by systems engineers when providing a service to the public. Overall, competence in ethics and professionalism can be summarized by a personal commitment to professional standards, recognizing obligations to society, the profession, and the environment.
Team Dynamics	Team dynamics are the unconscious, psychological forces that influence the direction of a team's behavior and performance. Team dynamics are created by the nature of the team's work, the personalities within the team, their working relationships with other people, and the environment in which the team works.
System Architecting	The definition of the system structure, interfaces, and associated derived requirements to produce a solution that can be implemented to enable a balanced and optimum result that considers all stakeholder requirements (business, technical….). This includes the early generation of potential system concepts that meet a set of needs and demonstration that one or more credible, feasible options exist.
Communications	The dynamic process of transmitting or exchanging information using various principles such as verbal, speech, body-language, signals, behavior, writing, audio, video, graphics, and language. Communication includes all interactions between individuals, individuals and groups, or between different groups.

Table 3. Candidate Recruitment Specific Proficiency Levels and Descriptions for a System Architect Supervised Practitioner Position

Competency	Proficiency Level	Description
Systems Thinking	Supervised Practitioner	Uses the principles of system partitioning within system hierarchy on a project.
Ethics and Professionalism	Supervised Practitioner	Acts ethically when fulfilling own responsibilities.
Team Dynamics	Supervised Practitioner	Uses team dynamics to improve their effectiveness in performing team goals.
Team Dynamics	Practitioner	Acts collaboratively with other teams to accomplish interdependent project or organizational goals.
System Architecting	Supervised Practitioner	Uses analysis techniques or principles used to support an architectural design process.
System Architecting	Supervised Practitioner	Develops multiple different architectural solutions (or parts thereof) meeting the same set of requirements to highlight different options available.
System Architecting	Supervised Practitioner	Produces traceability information linking differing architectural design solutions to requirements.
System Architecting	Supervised Practitioner	Uses different techniques to develop architectural solutions.
System Architecting	Supervised Practitioner	Prepares architectural design work products (or parts thereof) traceable to the requirements.
Communications	Practitioner	Uses appropriate communications techniques to ensure a shared understanding of information with all project stakeholders.
Communications	Practitioner	Uses appropriate communications techniques to ensure positive relationships are maintained.

Table 4. Candidate Recruitment Job Description

Job Overview	
Job Title	System Architect
Department	Systems Engineering
Location	Various
Salary	To be determined
General Job Description	
The incumbent creates and maintains architectural products throughout the lifecycle integrating hardware, software, and human elements; their processes; and related internal and external interfaces that meet user needs and optimize performance. Collaborates with a diverse group of systems engineers, project managers, engineers, and support staff.	
Duties and Responsibilities (Tasks)	
Technical • Uses the principles of system partitioning within system hierarchy on a project. • Uses analysis techniques or principles used to support an architectural design process. • Develops multiple different architectural solutions (or parts thereof) meeting the same set of requirements to highlight different options available. Develops multiple different architectural solutions (or parts thereof) meeting the same set of requirements to highlight different options available. • Produces traceability information linking differing architectural design solutions to requirements. • Uses different techniques to develop architectural solutions. • Prepares architectural design work products (or parts thereof) traceable to the requirements. **Ethics and Professionalism** • Acts ethically when fulfilling own responsibilities. **Teaming and Communications** • Uses team dynamics to improve their effectiveness in performing team goals. • Acts collaboratively with other teams to accomplish interdependent project or organizational goals. • Uses appropriate communications techniques to ensure a shared understanding of information with all project stakeholders. • Uses appropriate communications techniques to ensure positive relationships are maintained.	

define the expected competence that should be demonstrated for each respective level.

TAILORING COMPETENCIES

Competency frameworks, in general, are tailored so that organizations can develop competency models that are suited to their needs for their workforce. The framework is a good starting point to develop competency models. When tailoring to develop a specific instance of a systems engineering competency model, developers use sources other than the framework as necessary to provide adequate descriptions of the organization's roles.

HOW TO USE THE SYSTEMS ENGINEERING COMPETENCY FRAMEWORK: USAGE SCENARIOS

We present several usage scenarios to demonstrate how organizations use the framework and assessment guide in practice. All scenarios are based on those in (Presland, Whitcomb, and Zipes 2022). The scenarios are

- Job candidate recruitment
- Individual competency assessment
- Job description tailoring for an engineering job that needs systems engineering competencies
- Job description tailoring for a systems engineering job requiring an update for model-based systems engineering (MBSE) and digital engineering

For examples of the usage scenarios of frameworks, for training and education, as well as other examples, see (Presland, Whitcomb, and Zipes 2022) and (Whitcomb, Khan, and White 2016).

Candidate Recruitment

In this scenario, an organization needs to recruit an entry-level system architect. A job announcement targeting recent graduates from universities is created. Competency managers for the organization determine there are five desired competencies associated with the system architect job, listed in Table 2.

The list contains the system architecting competency and other competencies needed for the job. The individual writing the job description typically begins by filling out a company provided template using the descriptions of the five desired competencies for the specific proficiency levels desired. See the results in Table 3.

Note that the proficiency levels selected for any competency at a level desired for that specific competency and that the levels are not required to be the same across all competencies. Competencies exist at various proficiency levels, as these represent levels the competency managers desire, as well as being typical for recent university graduates. In this example, we take many competencies from the supervised practitioner level, and others from the practitioner level. We entered the competency and skill statements in the company template to develop the job description. See the results in Table 4.

We listed the task statements in multiple groupings to separate the specific technical

Table 5. Candidate Competency Assessment

Competency	Proficiency Level	KSAs (Can the employee do the following?)
System Architecting	Practitioner	Follows defined plans, processes, and associated tools to perform system architecting on a project. Recognizes situations where deviation from published plans and processes or clarification from others is appropriate to overcome complex system architecting challenges.
System Architecting	Practitioner	Develops alternative architectural design solutions from a set of requirements. Uses architectural frameworks in assisting consistency and re-usability of architectural design. Identifies the merits or consideration in different architectural design solutions. Uses an architectural design tool, methodology or modelling language. Uses different architectural approaches to establish preferred solution approaches of different stakeholders.
System Architecting	Practitioner	Describes the purpose and potential challenges of reviewing different architectural design solutions. Performs architecture trade-offs in terms of finding an acceptable balance between constraints such as performance, cost, and time parameters. Selects preferred options from those available, listing advantages and disadvantages.
System Architecting	Practitioner	Lists and describes key characteristics of different analysis techniques (cost analysis, technical risk analysis effectiveness analysis or other recognized formal analysis techniques). Uses techniques for analyzing the effectiveness of a particular architectural solution and selecting the most appropriate solution. Describes the advantages and limitations of the use of architectural design tools in relation to at least one tool.
System Architecting	Practitioner	Identifies areas where discipline implementation or technology constraints dictate partitioning of functionality (software, hardware, human factors, packaging, and safety. Develops architectural partitioning between discipline technologies.
System Architecting	Practitioner	Analyzes potential options against selection criteria. Selects credible solutions using criteria.

ones from the more general ones that might apply across most roles in the organization. An important aspect is that the framework provides a consistent ability to use competency and task statements for elements of a systems engineering job description, such that the person filling the job should be able to demonstrate the abilities listed. The job description would then be posted along with other desired information such as education required, location, salary compensation, and other job and employment information.

Individual Competency Assessment

In this scenario, suppose that an individual is currently at the Supervised Practitioner level for a required competency in System Architecting. The individual has already demonstrated behaviors for Supervised Practitioner, so the individual and supervisor concentrate on the assessment of achievement of Practitioner by reviewing the performance of tasks related to the System Architecting Practitioner, as shown in Table 5.

The assessment might include providing documented evidence for the respective knowledge, skill, ability, or behavior, or may require answering questions verbally as evidence of knowledge, or by demonstrating the knowledge, skill, ability, or behavior using a model-based systems engineering tool. Once the individual is verified as meeting the required knowledge, skill, ability, or behavior, the supervisor can provide their input to a record to verify the individual's level of achievement. Then, the employee is documented as having achieved the higher-level proficiency of a practitioner for system architecting. The supervisor must assess the employee for as many competencies and roles for the specialties and subspecialties as required for the individual.

The overall context for individual development is captured in a comprehensive career development plan, which is a career progression guide that will provide individuals with information about the types of skills and enrichment activities needed to further their career goals.

Job description tailoring for an engineering job

In this scenario, a competency manager would like to ensure that one of their engineering jobs require systems engineering competencies. In this case, a competency manager updates an existing materials engineer job description to include systems engineering related competencies and tasks. They start by retrieving the existing job description as shown in Table 6 (next page). For this example, the existing job description only uses a subset of the tasks required for an individual with a specialty in materials engineering.

Next, the competency manager reviews the job description to highlight any of the tasks that require systems engineering competencies. The following tasks from the existing job description in Table 6 were determined to be related to systems engineering tasks and the authors targeted the tasks to tailor with language from the framework and assessment guide.

Table 6. Existing Job Description Summary

Job Highlights
The incumbent develops and maintains an internationally recognized research program and leads a diverse group of scientists and support staff in research to develop new bio-based materials and value-added products from traditional as well as new or specialty sources.
Salary: To be determined
Location: Various
Qualifications
• BS degree in applicable Material Science / Material Engineering, Chemical Engineering, Manufacturing Engineering. Master's degree a plus. • 5+ years of Sheet Molding/Composite material development, material molding and leveling technology with proven results. • Strong understanding of batch material processing and process refinement. • Demonstrated critical thinking skills. Able to articulate issues across the organization. • Can effectively lead/work with other functions (Design Engineer, Manufacturing Engineer, Maintenance, Quality, and Distribution) for business results.
Responsibilities (Tasks)
• Develops and defines environmentally compliant material substitutes. • Solve complex problems and provide authoritative technical advice and/or instructions about material processes, materials failure analyses, and testing of materials for specification conformance. • Monitors the materials related to the industrial processes in the assigned areas by establishing and monitoring schedules, analysis procedures, and the qualifying criteria for the materials process monitored. • Provides expert advice and instructions concerning the materials aspect of processing or repair of materials, parts, and/or assemblies undergoing periodic rework, overhaul, repair, or modification on all systems. • Represent the company at conferences and meetings. Improve and maintain working relations with industry partners and actively participates in meetings to solve problems and contribute to special development projects. • Develop, assemble, and issue local engineering or process specifications consuming the materials aspects of processing and repair of materials, parts, and/or assemblies undergoing periodic rework, overhaul, repair, modification, as well as processing of new or experimental materials. • Review and evaluate correspondence related to material problems for technical accuracy and for development and action. • Present data in such a manner as to recognize and emphasize both theoretical and practical aspects. • Conduct engineering investigations. • Leads special material research and development projects.

Systems Thinking

- Solve complex problems and provide authoritative technical advice and/or instructions about material processes, materials failure analyses, and testing of materials for specification conformance.

Lifecycle Support

- Develop, assemble, and issue local engineering or process specifications consuming the materials aspects of processing and repair of materials, parts, and/or assemblies undergoing periodic rework, overhaul, repair, modification, as well as processing of new or experimental materials.

Quality

- Monitors the materials related to the industrial processes in the assigned areas of the aerospace rework facility serviced by establishing and monitoring schedules, analysis procedures, and the qualifying criteria for the materials process monitored.

Modeling and Analysis

- Leads special material research and development projects.

The competency manager then researches other systems engineering competencies and tasks that the organization would like to add to the job description. A list of additional key systems engineering competencies and tasks required for the job is complied and used to add systems engineering tasks to this Materials Engineer position are shown in Table 7.

Each of the tasks aligns with one or more competencies. The competency manager develops a final overall list of competencies that cover each task included for the position to include the systems engineering tasks. The framework and assessment guide identified key competencies required for the job based on the information. The framework and assessment guide identified a total of six competencies for the job.

- Systems Thinking
- Lifecycles
- Systems Modeling and Analysis
- Quality
- Capability Engineering
- General Engineering

Next, the competency manager uses the framework and assessment guide to find evidence indicators and organizes them as the basis for creating tailored tasks to update the job description, as shown in Table 8.

Throughout this process, the competency manager updates, rewrites, or adds new

Table 7. *Additional Tasks Desired for the New Job Description*

Task Descriptions
Develop holistic research and development of materials over a range of situations throughout the system lifecycle.
Ensure the adoption, introduction, and use of novel techniques and ideas related to quality initiatives across the enterprise and integrate materials engineering with enterprise-level quality initiatives, and create enterprise-level policies, guidance and best practice related to Materials Engineering quality for systems.
Prepare technology plans that address innovation, risk, maturity, readiness levels, and insertion points considering system capability.
Ensure development and performance across the enterprise are creative or innovative approaches to Materials Engineering.
Create overall processes for modeling and analysis tools for Materials Engineering that address systems aspects.

Table 8. *KSA Statements Related to the Tasks Desired for the New Job Description*

Competency	Proficiency Level	Indicators of Competence
Systems Thinking	Practitioner	Uses appropriate systems thinking approaches to a range of situations, integrating the outcomes to get a full understanding of the whole.
Lifecycles	Practitioner	Identifies dependencies aligning lifecycles and lifecycle stages of different system elements accordingly.
Systems Modeling and Analysis	Practitioner	Creates a governing process, plan and associated tools for systems modeling and analysis to monitor and control systems modeling and analysis activities on a system or system element.
Quality	Lead Practitioner	Promotes the introduction and use of novel techniques and ideas across the enterprise which improve the integration of Systems Engineering and quality management functions.
Quality	Lead Practitioner	Creates enterprise-level policies, procedures, guidance, and best practice in order to ensure Systems Engineering quality-related activities integrate with enterprise-level quality goals including associated tools.
Capability Engineering	Practitioner	Prepares technology plan that includes technology innovation, risk, maturity, readiness levels and insertion points into existing capability.
General Engineering	Lead Practitioner	Fosters creative or innovative approaches to performing general engineering activities across the enterprise.

tasks to create a comprehensive new job description. The plain text represents items from the original job description. The bold text represents portions from the framework and assessment guide competencies. Table 9 (next page) displays the updated example job description.

Job description tailoring updating a systems engineering job

In this scenario, an organization would like to update a systems engineering job description to include tasks related to model-based systems engineering and digital engineering competencies. A competency manager creates a new systems architect job description that includes MBSE and digital engineering related tasks. Table 10 (page 25) displays the current job description.

The organization's goal is to include relevant MBSE and digital engineering related tasks and competencies in the new job description. The competency manager starts with the existing job description task list. Next, they collect a list of key MBSE, and digital engineering tasks required for the job. These competencies and tasks are not always included in the framework or assessment guide. They use sources that document modern digital skills as a basis to tailor the competencies, such as SFIA (SFIA 2022). The source used for this example is the Digital Engineering Competency Framework (SERC 2021). The competency manager finds desired tasks and competencies formatted into the Competency Groups (G), Competency Subgroups (S), that address desired tasks. Table 11 (page 26) displays the DECF competency groups (SERC 2021).

Table 12 (page 27) displays the Competency Subgroups for the respective Competency Groups (SERC 2021).

The competency manager creates a final list of desired tasks at defined proficiency levels for MBSE and digital engineering.

C3 Modeling – Intermediate

- Create system models for system development efforts in accordance with applicable standards and policies
- Define the inter-relationships among model elements and diagrams
- Model current and desired scenarios, as directed
- Review the system model created by others
- Select appropriate tools and techniques for system modeling and analysis
- Use modeling language, concepts, diagrams, data attributes

C4 Simulation – Intermediate

- Applies modeling and simulation applications and tools, to cover a full range of modeling situations
- Integrate modeling capabilities with other product and analytical models including physics-based models
- Use simulation tools and techniques to

Table 9. The New Job Description

Job Highlights
The incumbent develops and maintains an internationally recognized research program and leads a diverse group of scientists and support staff in research to develop new bio-based materials and value-added products from traditional as well as new or specialty sources.
Salary: To be determined
Location: Various
Qualifications
• BS degree in applicable Material Science/Material Engineering, Chemical Engineering, Manufacturing Engineering. Master's degree a plus. • 5+ years of Sheet Molding/Composite material development, material molding and leveling technology with proven results. • Strong understanding of batch material processing and process refinement. • Demonstrated critical thinking skills. Able to articulate issues across the organization. • Can effectively lead/work with other functions (Design Engineer, Manufacturing Engineer, Maintenance, Quality, and Distribution) for business results.
Responsibilities (Tasks)
• Develops and defines environmentally compliant material substitutes. • **Uses appropriate systems thinking approaches to a range of situations, integrating the outcomes to get a full understanding of the whole** while providing authoritative technical advice and/or instructions regarding material processes, materials failure analyses, and testing of materials for specification conformance. • **Creates and assesses enterprise-level policies, procedures, guidance, and best practice to ensure** Materials Engineering **quality-related activities integrate with enterprise-level quality goals including associated tools.**. • **Promotes the introduction and use of novel techniques and ideas** across the enterprise **which improve the integration of Systems Engineering and quality management functions**. • Provides expert advice and instructions concerning the materials aspect of processing or repair of materials, parts, and/or assemblies undergoing periodic rework, overhaul, repair, or modification on all systems. • **Prepares** materials **technology plan that includes technology innovation, risk, maturity, readiness levels and insertion points into existing capability**. • Represents the company at conferences and meetings. Improve and maintain working relations with industry partners and actively participates in meetings to solve problems and contribute to special development projects. • Develops, assembles, and issues local engineering or process specifications consuming the materials aspects of processing and repair of materials, parts, and/or assemblies undergoing periodic rework, overhaul, repair, modification, as well as processing of new or experimental materials **identifying dependencies aligning lifecycles and lifecycle stages of different system elements accordingly**. • Reviews and evaluates correspondence related to material problems for technical accuracy and for development and action. • Presents data in such a manner as to recognize and emphasize both theoretical and practical aspects. • Conducts engineering investigations. • **Fosters creative or innovative approaches to performing** Materials Engineering **activities across the enterprise**. • **Creates a governing process, plan and associated tools** for support of research and development efforts **for systems modeling and analysis to monitor and control** materials **modeling and analysis activities on a system or system element**.

represent a system or system element

C8 Digital Architecting – Basic

- Utilize modeling languages to create or maintain system architectural products based on data provided
- Comply with style guides to properly develop architectural products

C8 Digital Architecting – Intermediate

- Provide architecture assessment to make decisions based on the architecture to ensure requirements are met for the system development throughout the lifecycle
- Apply system model and architectural concepts based on different stakeholder views and how they relate
- Collaborate with disciplinary subject matter experts to create system models and architectural products
- Create required architectural products for a system or system-of-systems in accordance with applicable standards and policies with minimal or no supervision

C11 Model-Based Systems Engineering Process – Basic

- Develop digital model artifacts, according to intent
- Analyze and interpret the results obtained using model-based engineering methods and tools

Table 10. The Current Job Description to be Tailored Through Adding Digital and MBSE

Job Overview	
Job Title	System Architect
Department	Systems Engineering
Location	Various
Salary	To be determined

General Job Description

The incumbent creates and maintains architectural products throughout the lifecycle integrating hardware, software, and human elements; their processes; and related internal and external interfaces that meet user needs and optimize performance. Collaborates with a diverse group of systems engineers, project managers, engineers, and support staff.

Duties and Responsibilities (Tasks)

Technical

- Develops an architecture model that includes functional and structural partitioning, interface definitions, design decisions, and requirements traceability.
- Develops or modifies architectures to meet organizational goals.
- Generates the full architectural design description for a system or program.
- Assesses the integrity of the overall architectural model to ensure that it meets the operational/system requirements and the business/mission needs.
- Develops various scenarios for system use, functions, and performance in the target environment.
- Collaborates with a disciplinary specialist to synthesize information across multiple users concerning the work tasks, work context, and socio-cultural factors that affect the system.
- Evaluates disciplinary specialist mitigations, designed by the disciplinary specialist, to gauge the effect on the operational use of the system.
- Determines the technical and programmatic areas where stakeholders reach agreements based upon the systems interfaces and interoperability concerns.
- Manages the creation of systems engineering architecture artifacts required as inputs to other functions.
- Partitions between discipline technologies to derive discipline specific requirements.
- Recommends partial architectural solutions that meet an important subset of needs quickly, coupled with approaches to seek better or more effective solutions.
- Reevaluates the solution design space and recommends adjustments when the solution does not meet customers' needs or fit the situation.
- Synthesizes individual solutions into larger solutions to explore innovative approaches.
- Identifies systems interfaces and interoperability concerns.

Management

- Document architecture structure in reports to assist in project reviews and decision making.

Communication

- Communicate effectively with disciplinary engineers to achieve system-level objectives.

Education and Training

Degree: Engineering or combination of education and experience. College-level education, training, and/or technical experience that furnished (1) a thorough knowledge of the physical and mathematical sciences underlying engineering, and (2) a good understanding, both theoretical and practical, of systems engineering and techniques and their applications.

Competencies

- System Architecting
- Modeling and Simulation
- Project Management
- Communications

Table 11. The DECF Digital and MBSE Competencies added

Group	Competency	Definition	Subgroup
G2	Modeling and Simulation	Use of digital models to describe and understand phenomena of interest from initiation of the effort through the entire life cycle maturation. Model literacy—understanding what models are and how they work—is required to move into more advanced skills, from the ability to build a model using appropriate tools, standards, and ontology to creating a modeling environment.	S2
G3	Digital Engineering and Analysis	Apply traditional engineering methods and processes in a digital environment. Create new engineering processes and methods for a digital environment. Create digital artifacts throughout the project or system lifecycle. Use engineering methods, processes, and tools to support the engineering and system lifecycle.	S3
			S4
G5	Digital Enterprise Environment	Addresses development of the Digital Engineering environment including hardware and software aspects. Digital Enterprise Environment Management is for management, communications and planning related to enabling the workforce to manage the adoption of appropriate model-based tools and approaches, techniques, and processes for the operation of digital enterprise environment systems that ensure transformational processes in enterprises occur with pace, high-quality and security. Digital Enterprise Environment Operations and Support within a digital enterprise environment include abilities to operate and support the digital enterprise environment across the enterprise and lifecycle. Digital Enterprise Environment Security involves developing policies, standards, processes, and guidelines to ensure the physical and electronic security of digital environments and automated systems.	S7

C11 Model-Based Systems Engineering Process – Intermediate

- Analyze the system model and architectural products
- Build models in a digital enterprise environment collaborative modeling environment
- Ensure digital artifacts are up-to-date, consistent, interoperable, accessible, uncorrupted, and properly and safely stored
- Generate digital enterprise environment system models
- Integrate all other model domains and physics-based models with the system model

C12 Digital Model-Based Reviews – Intermediate

- Conduct model-based reviews and audits, to ensure effective collaboration for system-of-interest evolution
- Confer with subject matter experts on models produced to gain concurrence on results
- Review resulting models with stakeholders and gain resolution to resultant issues

C20 Management – Intermediate

- Produce routine reports to assist in digital environment management activities and decision making

C21 Communication – Basic

- Communicate using digital model artifacts from the digital enterprise environment

The competency manager tailors and reorganizes the original tasks and competencies integrating them with appropriate indicators of competence from the framework and assessment guide. We used this information to create the updated job description, shown in Table 13 (page 28).

The plain text entries are based on the indicators of competence statements. The bold text portions come from the MBSE and digital engineering source statements. The italicized text are portions retained from the original job description.

SUMMARY

The framework and assessment guide should be quite useful in practice for any company or enterprise looking to gain value from their workforce organization. The usage scenarios demonstrate how the framework and assessment guide can be used. The four usage scenarios show how to tailor the framework and assessment guide for systems engineering applications. For further information on the framework and assessment guide, please refer to the source publication (Presland, Whitcomb, and Zipes 2022). ■

REFERENCES

- Beasley, R., Gelosh, D.S., and Pickard, A.C. 2019. “Professional Competencies – the soft skills to give Systems Engineers a hard edge.” *INCOSE International Symposium* 29 (1): 301-317. https://doi.org/10.1002/j.2334-5837.2019.00605.x
- Gelosh, D.S., Heisey, M., Snoderly, J.R., and Nidiffer, K. 2016. “The Path to Version 0.75 of the Proposed INCOSE Systems Engineering Competency Framework.” *INCOSE International Symposium* 26 (1): 1852-1863. https://doi.org/10.1002/j.2334-5837.2016.00265.x
- Gelosh, D.S., Snoderly, J.R., Heisey, M., and Nidiffer, K. 2015. “Version 0.5 of the Proposed INCOSE Systems Engineering Competency Framework.” *INCOSE International Symposium* 25 (1): 112-138. https://doi.org/10.1002/j.2334-5837.2015.00052.x
- Gelosh, D.S., Heisey, M., Snoderly, J.R., and Nidiffer, K. 2017. “Version 0.75 of the Proposed INCOSE *Competency Framework*.” *INCOSE International Symposium* 27 (1): 45-56. https://doi.org/10.1002/j.2334-5837.2017.00344.x
- Gelosh, D.S., Snoderly, J.R., Heisey, M., Anthony, J.F., Jr., and Nidiffer, K. 2014. “7.2.2 Developing the Next Generation of the INCOSE Systems Engineering Competency Framework.” *INCOSE International Symposium* 24 (1): 635-642. https://doi.org/10.1002/j.2334-5837.2014.tb03172.x

Table 12. The Added Details for the DECF Digital and MBSE Competencies

Subgroup/Title	Number/Title	Definition
S2 Modeling and Simulation	C3 Modeling	Modeling is essential to aid in understanding complex systems and system interdependencies, and to communicate among team members and stakeholders.
	C4 Simulation	Simulation provides a means to explore concepts, system characteristics, and alternatives; open the trade space; facilitate informed decisions and assess overall system performance.
S3 Digital Systems Engineering	C8 Digital Architecting	Digital architecture activities use digital models to define a comprehensive digital system model based on principles, concepts, and properties logically related to and consistent with each other. Digital architecture has features, properties, and characteristics that satisfy, as far as possible, the problem or opportunity expressed by a set of system requirements (traceable to mission/business and stakeholder requirements) and life cycle concepts (operational, support) and which are implementable through digital enterprise related technologies. Digital architecture competencies relate to the ability to create system digital models and required architectural products and digital artifacts for a system or system-of-systems in accordance with applicable standards and policies.
	C11 Model-Based Systems Engineering Processes	Model-based systems engineering is the formalized application of modeling to support system requirements, design, analysis, verification, and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases.
S4 Engineering Management	C12 Digital Model-Based Reviews	Digital model-based reviews define the series and sequence of model-based systems engineering activities that bring stakeholders to the required level of commitment, prior to formal reviews. It utilizes system models, artifacts, and products for analysis of design and technical reviews to execute trade-off and design analyses, prototyping, manufacturing, testing, and sustainment of the system.
S7 Digital Enterprise Environment Management	C20 Management	Management in the digital enterprise environment aims to deliver a framework that ensures transformational processes in enterprises occur with pace, high-quality, and security. This is achieved through a set of IT solutions that designed to make digital businesses fast, seamless, and optimized at every level.
	C21 Communications	Communications include using digital model artifacts from the digital enterprise environment to investigate and manage the adoption of appropriate model-based tools, techniques, and processes for the operation of digital enterprise environment systems and services. Communications also establishes the appropriate guidance to enable transparent decision-making, allowing senior leaders to ensure the needs of principal stakeholders are understood, the value proposition offered by digital enterprise environment is accepted by stakeholders, and the evolving needs of the stakeholders and their need for balancing benefits, opportunities, costs, and risks is embedded into strategic and operational plans.

- Gelosh, D.S., Heisey, M., Nidiffer, K., Snoderly, J.R., and Beasley, R. 2018. "Version 1.0 of the New INCOSE Competency Framework." *INCOSE International Symposium* 28 (1): 1778-1786. https://doi.org/10.1002/j.2334-5837.2018.00583.x
- Holt, J., and Perry, S. 2011. *A Pragmatic Guide to Competency, Tools, Frameworks, and Assessment*. Swindon, ENG-UK: BCS.
- SERC (Systems Engineering Research Center) Technical Report SERC-2021-TR-005. 2021. Developing the Digital Engineering Competency Framework (DECF) – Phase 2. Hoboken, US-NJ: Stevens Institute of Technology.
- IEEE (Institute of Electrical and Electronics Engineers). 2022. IEEE P1484.20.2:2022. Recommended Practices for Defining Competencies. Geneva, CH: IEEE.
- INCOSE UK. 2010. "SE Competencies Framework Issue 3." https://incoseuk.org/Documents/Groups/UKChapter/SE_Competencies_Framework_Issue_3.pdf.
- Presland, I., ed. 2018. Systems Engineering Competency Framework. INCOSE-TP-2018-002-01.0.
- Presland, I., Whitcomb, C., and Zipes, L. 2022. *The Systems Engineering Competency Assessment Guide*. Hoboken, US-NJ: Wiley.
- SFIA. 2022. "The global skills and competency framework for the digital world". https://sfia-online.org/en.

Table 13. The New Job Description

Job Overview	
Job Title	System Architect
Department	Systems Engineering
Location	Various
Salary	To be determined
General Job Description	
The incumbent develops and maintains system architectures using model-based methods and collaborates with a diverse group of systems engineers, project managers, disciplinary engineers, and support staff to develop system models using digital modeling tools and environments.	
Duties and Responsibilities (Tasks)	
Technical • Creates alternative architectural designs **in a digital enterprise environment modeling environment** traceable to the requirements **in accordance with applicable standards and policies** to demonstrate different approaches to the solution **and reviews resulting models with stakeholders and gains resolution to resultant issues**. • Uses appropriate analysis techniques to consider different viewpoints **based on different stakeholder views and how they relate**. • Elicits derived discipline specific architectural constraints from specialists to support partitioning and decomposition, **synthesizes individual solutions into larger solutions to explore innovative approaches and evaluates disciplinary specialist mitigations, designed by the disciplinary specialist, to gauge the effect on the operational use of the system**. • Identifies *systems interfaces and interoperability concerns* and the impact on interface definitions because of wider changes. • Defines **and utilizes** appropriate **modeling language, concepts, diagrams, data attributes, and** representations of a system or system element **to create or maintain system architectural products based on data provided**. • Selects appropriate tools and techniques for system modeling and analysis. • Analyzes options and concepts **using model-based engineering methods and tools and integrates all other model domains and physics-based models with the system model** to demonstrate that credible, feasible options exist. • Uses the results of system analysis activities to *reevaluate the solution design space* and inform system architectural design *and recommend adjustments when the solution does not meet customers' needs or fit the situation*. • Monitors key aspects of the evolving design solution *and recommends partial architectural solutions that meet an important subset of needs quickly, coupled with approaches to seek better or more effective solutions* in order to adjust architecture, if appropriate. **Management** • Follows governing project management plans, and processes and uses appropriate **digital environment management** tools to control and monitor project management-related Systems Engineering tasks, interpreting, as necessary. • Develops **digital model artifacts, according to intent, and ensures digital artifacts are up-to-date, consistent, interoperable, accessible, uncorrupted, and properly and safely stored for use in model-based reviews and audits** as Systems Engineering inputs for project management status reviews to enable informed decision-making. **Communication** • Uses appropriate communications techniques **using digital model artifacts from the digital enterprise environment** to ensure a shared understanding of information with all project stakeholders. • Uses appropriate communications techniques to express alternate points of view in a diplomatic manner using the appropriate means of communication **to collaborate with disciplinary subject matter experts to create system models and architectural products and gain concurrence on results**.	
EDUCATION AND TRAINING	
Degree: Engineering or combination of education and experience. College-level education, training, and/or technical experience that furnished (1) a thorough knowledge of the physical and mathematical sciences underlying engineering, and (2) a good understanding, both theoretical and practical, of the engineering sciences and techniques and their applications to one of the branches of engineering.	
COMPETENCIES	
• System Architecting • Systems Modeling and Analysis • Interfaces • Project Management • Communications	

- Whitcomb, C.A., Khan, R., and White, C. 2016. "Curriculum Alignment Use Case for Competency Frameworks at the Naval Postgraduate School." *INCOSE International Symposium* 26 (1): 105-114. https://doi.org/10.1002/j.2334-5837.2016.00148.x
- Whitcomb, C., Khan, R., Grambow, D., Velez, J., Delgado, J., and White, C. 2017. "A Description of the Defense Systems Engineering Career Competency Model." Proceedings of the Fourteenth Annual Acquisition Research Symposium, Monterey, US-CA, 26-27 April.

ABOUT THE AUTHORS

Clifford Whitcomb, PhD is a Professor in the Systems Engineering Program at Cornell University. He was previously a Distinguished Professor in the Systems Engineering Department at the Naval Postgraduate School in Monterey, CA. He is an INCOSE Fellow and a Fellow of the Society of Naval Architects and Marine Engineers (SNAME). He supports several competency framework and model development efforts. He is the Chair of the INCOSE Competency Working Group. He was a principal investigator for the development of the Systems Engineering Career Competency Model (SECCM) now formally a part of the US Navy Human Resources Competency Network. He collaborated as part of the team that developed the Digital Engineering Competency Framework (DECF) with the SERC. He is a member of the IEEE working group developing the initial IEEE P1484.20.2:2022 Recommended Practices for Defining Competencies and an INCOSE liaison for the ISO/IEC JTC 1/SC 7/WG 20 "Software and systems bodies of knowledge and professionalization."

Corina L. White is a Lecturer in the Systems Engineering Department at the Naval Postgraduate School (NPS), Monterey, California. Her expertise is in over three distinct disciplines including research and development, depot maintenance, force protection, DoD acquisition, and test and evaluation. She has had the opportunity to work with the National Aeronautics and Space Administration (NASA), Naval Air Systems Command (NAVAIR), Marine Corps Test and Evaluation Activity (MCOTEA), US Air Force Lifecycle Management Center (AFLCMC) and currently the Naval Postgraduate School (NPS). She graduated with a bachelor's in chemical engineering from Prairie View A&M University in 2007 and earned a master's in systems engineering in 2014 from the Naval Postgraduate School. She is ENG Level III certified and has a Digital Engineering certificate from DAU. Ms. White's research experience includes the development of polymeric materials to improve radiation-shielding capabilities for astronauts, bio-catalytic transformations, assisting with the creation of a Systems Engineering Career Competency Model (SECCM) to aid in the development of position descriptions and career development plans for systems engineers within the Department of the Navy and aiding in the development of an ontology and conceptual data model for the United States Marine Corps in support of the Digital Engineering Transformation efforts.

Rabia H. Khan is a Faculty Research Associate in the Systems Engineering Department at the Naval Postgraduate School (NPS), Monterey, California. Ms. Khan has experience in RDT&E, involving projects related to systems engineering, defense energy, modeling & simulation, and operations research. She earned her bachelor's degree in Neurobiology, Physiology and Behavior from the University of California, Davis and a master's degree in engineering systems from the Naval Postgraduate School. Ms. Khan's research experience includes assisting in the creation of the Systems Engineering Career Competency Model (SECCM), which aids in the development of position descriptions and career development plans for systems engineers within the Department of the Navy. She assisted in the development of the Digital Engineering Competency Framework (DECF) with the SERC.

McDermott and Hutchison continued from page 16

ABOUT THE AUTHORS

Tom McDermott serves as the Deputy Director and Chief Technology Officer of the Systems Engineering Research Center (SERC) at Stevens Institute of Technology in Hoboken, NJ. The SERC is a University Affiliated Research Center sponsored by the Office of the Secretary of Defense for Research and Engineering. With the SERC he develops new research strategies and is leading research on Digital Engineering transformation, education, security, and artificial intelligence applications. Mr. McDermott also teaches system architecture concepts, systems thinking and decision making, and engineering leadership. He consults with several organizations on enterprise modeling for transformational change, and often serves as a systems engineering expert on government major program reviews. He currently serves on the INCOSE Board of Directors as Director of Strategic Integration.

Dr. Nicole Hutchison is a Principal Investigator (PI) and research engineer at the Systems Engineering Research Center (SERC). Her primary work through the SERC has been in human capital development research. This has included the development of competency frameworks for systems engineering (the Helix project), digital engineering, and mission engineering. Currently, Dr. Hutchison is the PI for the Simulation Training Environment for Digital Engineering (STEDE), a project that is developing realistic models to be used in training the DoD acquisition workforce in a way that builds hands-on digital engineering skills. She previously served on the BKCASE research team, which resulted in the development of the *Guide to the Systems Engineering Body of Knowledge* (SEBoK). She is currently the Managing Editor for the SEBoK and the Lead Editor for the "Enabling Systems Engineering" section of the SEBoK. Before joining the SERC, she spent 5 years working for Analytic Services, Inc., supporting the US Departments of Defense, Homeland Security, Health and Human Services, and Justice. She holds a PhD in systems engineering from Stevens and is a Certified Systems Engineering Professional (CSEP) through INCOSE.

A Vision for Universal and Standardized Access to Systems Competency Education

Caitlyn A. K. Singam, caitlyn.singam@incose.net

ABSTRACT

The interdisciplinary nature of systems engineering inherently means that the concept of a systems engineer is variable, requiring several types of expertise depending on the application sector and knowledge domain the individual is working in. However, regardless of industry or specialty, systems engineers are all unified by proficiency in a core set of fundamental competencies that allow them to understand and holistically work with systems.

These systems thinking and systems engineering competencies are applicable by being interdisciplinary and are well-suited to an increasingly globalized and interconnected world. However, views of systems thinking and systems engineering competencies as non-essential supplementary skills rather than foundational ones have meant that systems thinking and systems engineering concepts are not generally included in science, technology, engineering, arts, and mathematics (STEAM) curricula until late in the educational pathway and are not always presented in a standardized manner. Consequently, under the current approach to systems thinking and systems engineering, not all individuals who can benefit from systems thinking and systems engineering education have access to it. Those who do often face substantial variability in the quality of their educational experiences.

This article discusses existing issues relating to systems thinking and systems engineering competency education, as well as the efforts of the INCOSE SySTEAM Initiative towards addressing those issues and facilitating the integration of systems thinking and systems engineering competencies into STEAM curricula at all levels of education. Furthermore, this article presents a guiding philosophy for promoting universal and standardized access to systems competency education.

1. INTRODUCTION: THE SIGNIFICANCE OF SYSTEMS SKILLS

To ask what defines a systems engineer – or, more broadly, a systems thinker – is to ask what might unite an artist with an architect, a rocket scientist with a political scientist, or a programmer with a philosopher. Different as these individuals may be, and regardless of whether they carry the formal title of 'systems engineer,' they are all unified by a common ability to understand and tackle problems from a holistic, integrated (systems-level) perspective: the systems thinking and system engineering skillset.

The premium placed on these systems thinking and systems engineering competencies is one of the main traits that set systems engineering practitioners, and indeed systems engineering as a field, apart from other technical disciplines. Unlike other technical fields, which require their practitioners to have mastery of specific, specialized knowledge within a narrow domain, systems engineering is inherently interdisciplinary (Kossiakoff et al. 2011). Rather than limiting its practitioners to one particular area of expertise, systems engineering equips individuals with a formalized 'systems approach' to problem-solving, encompassing standard skillsets identified across various disciplines. These competencies, formally documented and standardized by organizations such as the International Council on Systems Engineering (INCOSE), apply to many application areas. Systems engineers learn to recognize that they can construe any entity or idea in terms of interconnected and interacting systems: statutes in the legal system, components in an engineering project, schools in a district, cells in an organism, or even people in a community can all be couched in terms of systems and system components. Through that lens, any problem can be framed as a systems engineering problem, and systems engineers formally trained in systems thinking and systems engineering competencies can use those skills to streamline how they gain an understanding of and subsequently tackle those challenges (Arnold and Wade 2017).

The broad applicability of systems thinking and systems engineering competencies also means that individuals do not need to

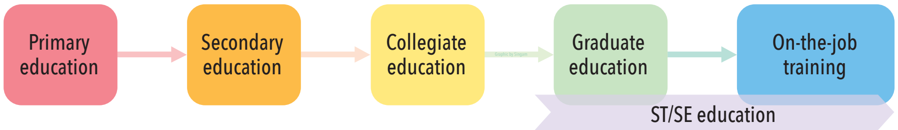


Figure 1. The current approach to systems thinking and systems engineering competency education

work in engineering to benefit from having systems thinking and systems engineering competencies. Individuals can apply the systems engineering process to many application areas or sectors, including those outside the engineering world. Today's interconnected and globalized world has highlighted the need for systems thinking and systems engineering skills such as interdisciplinarity (Er Saw and Jiang 2020; Taebi et al. 2014)closely connected to stakeholder dynamics. Hence, an ideal approach to responsible innovation requires interdisciplinary research that incorporates: (i, and an understanding of the complex interplay of interacting systems on a large scale (Newberry 2005) across a variety of disciplines and careers in the sciences and humanities alike. On a more fundamental level, systems thinking and systems engineering competencies equip individuals with skills such as leadership, teamwork, and innovation, often cited as critical to workplace readiness (Care et al. 2017).

2. CURRENT STATE OF SYSTEMS EDUCATION

Given the benefits of promoting systems thinking and systems engineering competency and its widespread applicability, it is surprising that systems thinking and systems engineering competencies are not presently integral to every student's education. The systems engineering community, after all, has been aware of the broad utility of systems thinking and systems engineering competencies essentially since the inception of systems engineering as a formal discipline (Schlager 1956), and has in recent years made several strides forward in setting standards for competency attainment. In addition to maintaining a competency framework (Gelosh et al. 2018), INCOSE maintains a certification program (Wright 2015; Walter and Walden 2010) that recognizes candidates' attainment of the knowledge and experience associated with systems thinking and systems engineering competencies. Similarly, the Accreditation Board for Engineering and Technology (ABET) recently approved program criteria for accrediting college and graduate programs in systems engineering (INCOSE 2021).

Despite these strides forward, systems thinking and systems engineering competencies are not generally recognized by the academic community as being integral portions of science, technology, engineering, arts, and mathematics (STEAM) curricula, especially in pre-collegiate education (Asbjornsen and Hamann 2000). Systems thinking and systems engineering education is instead typically treated as a supplemental, rather than foundational, training (Fig. 1) and is typically offered in the form of on-the-job professional development (Carlson 1991) or through dedicated graduate education programs in supplement to a prior technical degree in a different discipline (Muller 2005). While this approach is effective in addressing the direct pressures from industry for ensuring a basic level of competency among the systems engineering workforce and certainly helped grow the systems engineering community to where it is today, it does have a significant drawback of limiting the visibility of systems thinking and systems engineering education to individuals outside of the systems engineering community. The current approach does not guarantee that students get an opportunity to receive exposure to systems thinking and systems engineering competencies during their foundational education. Due to the small size of the systems engineering community, they are also often unaware of the existence of systems engineering as a field and career option. The absence of systems thinking and systems engineering in early education, combined with the lack of systems thinking and systems engineering content outside of engineering classrooms, thus means that many students therefore miss the chance to explore a potential interest in SE until after they have already specialized, and often never get to see those skills applied outside of an engineering context. It begets a vicious cycle in which future STEAM educators and academics outside of the systems engineering community are not exposed to systems thinking and systems engineering during their education, and thus omit systems thinking and systems engineering competencies from their curricula, leading to their students being unaware of systems thinking and systems engineering and continuing the pattern into the future.

The lack of awareness of systems thinking and systems engineering competencies and competency standards among the general, non-specialist educators has also led to various issues in terms of if and how systems thinking and systems engineering gets presented to students. It is well-documented in the literature that the quality and characteristics of the educational experience – even the portions governed by curricular standards – are known to vary greatly across disciplines, geographic regions, and even from student to student, with the implementation of curricula being impacted by factors such as resource availability (Augenblick, Myers, and Anderson 1997) and educator training (Verma 2021). In the case of something like systems thinking and systems engineering education, the lack of standardized guidelines means that there is far more variability. Thus, while some districts may be seeking to formally introduce systems thinking and systems engineering education in secondary school classrooms (Cerovac, Seemann, and Keane 2018), students in other districts might only get exposed to systems thinking and systems engineering prior to college if appropriate extracurriculars are available at their school (with the onus being placed on teachers and mentors to decide how to link systems thinking and systems engineering competencies to the activities being done), while others still may never get introduced to systems thinking and systems engineering at all.

This variability in systems thinking and systems engineering educational quality produces students with varying levels of skill in the core systems thinking and systems engineering competencies, with employers having to expend resources towards on-the-job training to address any gaps between a new hire's starting level of competency and the organization's needs. Although the existence of standardized competency frameworks as well as initiatives such as the INCOSE certification program do offer employers standardized benchmarks against which to evaluate job candidates, variability in educational standards has largely meant that employers do not consistently accept systems engineering degrees or educational qualifications when considering candidates, and instead opt for assessing candidates based on their on-the-job training and prior work experience. This matter is further complicated by job titles, which sometimes conflate general engineering roles involving specific types of systems ('instrument-system engineer') with individuals performing interdisciplinary systems-engineering work on a specific project ('instrument systems-engineer') and again add further variability to assessment of systems thinking and systems engineering experience and to the hiring process.

3. THE NEED FOR UNIVERSAL, STANDARDIZED SYSTEMS EDUCATION

Inconsistent implementation of sys-

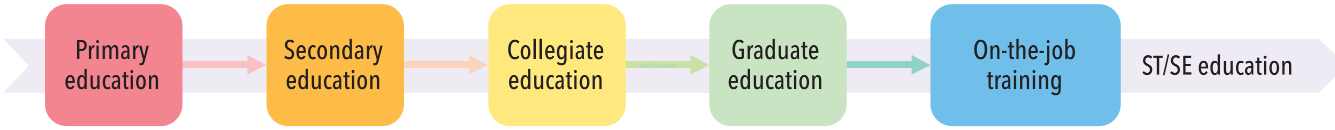


Figure 2. The integrated approach to continuous systems thinking and systems engineering education

Table 1. SySTEAM's vision for ST/SE integration across the educational pathway

Education level	systems thinking and systems engineering integration objective
Primary	Introduce students to the holistic systems-thinking perspective
Secondary	Promote systems thinking and systems engineering skills across disciplines (including arts) and introduce systems engineering as a potential career choice
Collegiate/university	Equip non-engineering majors with systems thinking and systems engineering skills relevant to their domains and provide engineering majors with interdisciplinary education & industry-relevant technical skills
Graduate education	Equip industry professionals with state-of-the-art systems thinking and systems engineering skills and knowledge so that they can advance the field and practice of systems thinking and systems engineering
Professional development	Provide professionals with the competencies needed for employers to hire quality employees and to succeed in the modern workforce

tems thinking and systems engineering competencies throughout the educational pathway—both in terms of educational access and quality—clearly creates visibility and variability problems which have a substantial impact on who learns about, and is given the opportunity to utilize, systems thinking and systems engineering skills.

Under the current systems thinking and systems engineering educational paradigm, individuals in the humanities and other fields outside of the systems engineering community can rarely get formal exposure to systems thinking and systems engineering outside of an engineering classroom despite potentially benefiting from taking a systems approach to their work. Those who do have an interest in engineering may still miss the opportunity to obtain systems thinking and systems engineering skills due to a lack of awareness about systems engineering as a field and career option, and early-career individuals who do become systems engineers may encounter difficulties in the hiring process due to inconsistent recognition of systems thinking and systems engineering competencies by various employers. It is a situation which results in the loss of potential talent that the workforce could otherwise benefit from, both in terms of reducing the number of individuals with systems thinking and systems engineering competencies and in limiting the diversity of individuals with those competencies.

4. THE INCOSE SYSTEAM VISION

There is clearly an acute need to address the matter of systems thinking and systems engineering competency education, specifically by increasing the accessibility and standardization of systems thinking and systems engineering education throughout the educational pathway.

To this end, the INCOSE SySTEAM Initiative (incose.org/systeam), an international volunteer-led effort, is working towards realizing a vision for integrating SE/ST into every STEAM classroom, with the goal of "improving STEAM education everywhere, for everyone." The SySTEAM mission is to improve the quality of STEAM education worldwide, for all students, by changing the way in which educators, administrators, and other relevant stakeholders place value on and leverage systems thinking and systems engineering skills, and by establishing a community to advocate for holistic integration of ST and SE principles and skills (inclusive of the skills represented in the INCOSE Competency Framework) into existing STEAM curricula and programs.

Specifically, SySTEAM aims to establish a framework for replacing the current 'supplemental training'-approach to systems thinking and systems engineering education (Fig. 1) with an integrated approach to systems thinking and systems engineering education across the entire educational pathway (Fig. 2). Rather than only introducing systems thinking and systems engineering skills as a type of specialty training, SySTEAM seeks to promote the perception of systems thinking and systems engineering competencies as foundational skills taught via a continuous learning process, with integration objectives tailored for each stage of the educational pathway (Table 1).

As part of its framework, SySTEAM seeks to generate recommendations for systems thinking and systems engineering education that are universally applicable across the following parameters:

- *Across the globe:* to meet the needs of a globalized workforce, there needs to be a baseline level of consistency in how systems thinking and systems engineering education is approached across the globe so that individuals can be eligible to work anywhere, regardless of where those individuals were educated.
- *Across all educational levels:* education should present systems thinking and systems engineering competencies as foundational skills at the earliest levels of education so that students can learn to use systems thinking and systems engineering competencies as tools for problem solving and can acclimatize to the holistic systems-thinking perspective.
- *Across both technical and non-technical fields:* all students need to be given the opportunity to benefit from the workforce-readiness skills provided by systems thinking and systems engineering competencies, and students in both the sciences and humanities alike should have a chance to explore how systems thinking and systems engineering can be utilized in various application domains.
- *Across all types of schools:* to be successful, recommendations for

Table 2. SySTEAM strategic principles

Principle	Framework development strategy
Accessibility & transparency	• Recommendations are accessible, applicable, and implementable by individuals inside and outside the systems engineering community • Introduce cross-cutting systems thinking and systems engineering skills earlier than technical concepts such as, requirements management
Inclusivity	• Recommendations should not require specialized resources or support • Recommendations cannot be exclusionary to, or highly impractical for, stakeholders outside of the engineering or STEM sector • Recommendations should reflect the relevance of SE/ST competencies to non-engineering fields and vice versa • Recommendations should ensure students should not have to be in an engineering classroom to get exposure to systems thinking and systems engineering competencies • Recommendations should align with international educational standards
Accountability	• Successful systems thinking and systems engineering integration should be verifiable via specific metrics
Quality- and action-driven performance	• Recommendations should be sustainable • Support recommendations with impact/risk assessments • Align recommendations with input from stakeholders
Integrity & ethical conduct	• Successfully implemented recommendations should not have an elevated risk of causing or exacerbating disproportional educational outcomes or otherwise negatively impacting educational performance

systems thinking and systems engineering integration need to be implementable by teachers in educational institutions with varying levels of resource access (recommendations for project-based learning should not require that students have access to specialized labs or facilities).

5. STRATEGIC PRINCIPLES FOR FRAMEWORK DEVELOPMENT

In developing a framework for systems thinking and systems engineering integration, SySTEAM has identified several key strategic principles which are necessary for guiding successful framework development. Table 2 addresses those strategic principles and what they entail with regards to systems thinking and systems engineering competency integration.

The framework uses these principles to guide SySTEAM's long-term efforts as it works towards a consolidated framework for integrating systems thinking and systems engineering with STEAM curricula. Successful frameworks meet the following requirements:

1. Implementable (with sufficient detail to be executed by a third party) recommendations for integrating systems thinking and systems engineering into STEAM education
2. Have a basis in community input;
3. Provide descriptions of the minimum resources and infrastructure required for implementation of the recommendations;
4. Provide recommendations addressing all the core competencies in the INCOSE Competency Framework;
5. Provide recommendations which span the entire STEAM educational pathway (primary to postgraduate);
6. Provide recommendations sourced from the common stakeholder needs identified from SySTEAM participant input (the framework should pass verification testing and be traceable to the requirements imposed on SySTEAM in this document;)
7. Include at least one section with recommendations specifically addressing the inclusion and integration of systems thinking and systems engineering skills in arts and humanities curricula, extending beyond simply mandating the inclusion of science/engineering coursework as a general education requirement;
8. Include commentary addressing recommended practices for implementing the framework in at least three different international regions/educational systems;
9. Provide an impact and risk assessment of the burden and repercussions expected with implementation of the recommendations;
10. Include appropriate commentary or auxiliary documentation providing the rationale for the recommendations provided, including references to existing studies, analyses, and standards as needed to justify the necessity or utility of such recommendations;
11. Recommend standardized methods for evaluating successful implementation of each recommendation via qualitative and quantitative means;
12. Receive approval from the academic, industry, and non-SE sectors, with at least three different approving stakeholder organizations representing each sector (the framework should pass validation testing;)
13. A copy posted online for public viewing and use, free of charge.

6. CONCLUSIONS

Every student can benefit from having systems thinking and systems engineering competencies; however, up until now the systems engineering community has neglected providing these skills with the time they deserve. The INCOSE SySTEAM Initiative presents a vision for addressing current issues with limited access to systems thinking and systems engineering competency education and variations in educational quality by embedding systems thinking and systems engineering competencies in every classroom and standardizing competency attainment expectations across the educational pathway. In support of that vision, it has established a strategy for developing a framework which can suitably describe a continuous-education model for systems thinking and systems engineering education whilst also remaining widely implementable on an international level and across all levels of education.

With the support of its volunteer community and its strategic infrastructure, SySTEAM is ready to develop a framework that will allow every student to count systems thinking and systems engineering competencies as part of their educational toolkit. ■

REFERENCES

- Arnold, R., and Wade, J. 2017. "A Complete Set of Systems Thinking Skills." *INSIGHT* 20 (3): 9–17. https://doi.org/10.1002/inst.12159
- Asbjornsen, O., and Hamann, R. 2000. "Toward a Unified Systems Engineering Education." *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* 30 (2): 175–82. https://doi.org/10.1109/5326.868438
- Augenblick, J., Myers, J., and Anderson, A. 1997. "Equity and Adequacy in School Funding." *The Future of Children* 7 (3): 63–78. https://doi.org/10.2307/1602446
- Care, E., Kim, H., Anderson, K., and Gustafsson-Wright, E. 2017. *Skills for a Changing World: National Perspectives and the Global Movement. Center for Universal Education at The Brookings Institution*. Center for Universal Education at The Brookings Institution. https://eric.ed.gov/?id=ED583031
- Carlson, H. 1991. "Systems Engineering Training Program." *INCOSE International Symposium* 1 (1): 50–52. https://doi.org/10.1002/j.2334-5837.1991.tb01482.x
- Cerovac, M., Seemann, K., and Keane, K. 2018. "Systems Engineering: Identification and Fostering." Paper presented at the 10th Biennial International Design and Technology, Hawthorn, ASTL-Vic, 5-8 December.
- Er Saw, P., and Jiang, S. 2020. "The Significance of Interdisciplinary Integration in Academic Research and Application." *BIO Integration* 1 (1): 2–5. https://doi.org/10.15212/bioi-2020-0005
- Gelosh, D., Heisey, M., Nidiffer, K., Snoderly, J., and Beasley, R. 2018. "Version 1.0 of the New INCOSE Competency Framework." *INCOSE International Symposium* 28 (1): 1778–86. https://doi.org/10.1002/j.2334-5837.2018.00583.x
- INCOSE. 2021. "ABET Engineering Area Delegation and Board of Delegates Approves the Systems Engineering Program Criteria." https://www.incose.org/docs/default-source/default-document-library/abet-acceptance-2021_final.pdf?sfvrsn=408b63c7_0.
- Kossiakoff, A., Sweet, W., Seymour, S., and Biemer, S. 2011. *Systems Engineering Principles and Practice*. Hoboken, US-NJ: John Wiley & Sons.
- Muller, G. 2005. "5.3.2 Didactic Recommendations for Education in Systems Engineering." *INCOSE International Symposium* 15 (1): 768–76. https://doi.org/10.1002/j.2334-5837.2005.tb00707.x
- Newberry, B. 2005. "Engineering Globalization: Oxymoron or Opportunity?" *IEEE Technology and Society Magazine* 24 (3): 8–15. https://doi.org/10.1109/MTAS.2005.1507535
- Schlager, K. 1956. "Systems Engineering-Key to Modern Development." *IRE Transactions on Engineering Management* EM-3 (3): 64–66. https://doi.org/10.1109/IRET-EM.1956.5007383
- Taebi, B., Correljé, A., Cuppen, E., Dignum, M., and Pesch, U. 2014. "Responsible Innovation as an Endorsement of Public Values: The Need for Interdisciplinary Research." *Journal of Responsible Innovation* 1 (1): 118–24. https://doi.org/10.1080/23299460.2014.882072
- Verma, G. 2021. *Inequality and Teacher Education: An International Perspective*. Milton Park, UK: Routledge.
- Walter, S., and Walden, D. 2010. "Associate Systems Engineering Professional (ASEP) Certification: A Credential Tailored For Students And Junior Engineers." Paper presented at the 2010 Annual Conference and Exposition, Louisville, US-KY, 20-23 July.
- Wright, C. 2015. "INCOSE Certification Program Knowledge Exam Update 2015." *INCOSE International Symposium* 25 (1): 93–111. https://doi.org/10.1002/j.2334-5837.2015.00051.x

ABOUT THE AUTHOR

Ms. Caitlyn Singam is an INCOSE-accredited systems engineer whose research interests include astrobiology, autonomous space systems, and risk mitigation via MBSE. She currently serves as the Program Director and initiative lead for the INCOSE SySTEAM Initiative and as an Assistant Editor for the Systems Engineering Body of Knowledge (SEBoK). She is a U.S. National Science Foundation (NSF) Fellow and member of Johns Hopkins University's Study of Exceptional Talent and holds a B.Sc. in biological sciences and an M.Sc. in systems engineering, both from the University of Maryland, College Park.

The Digital Engineering Competency Framework (DECF): Critical Skillsets to Support Digital Transformation

Nicole Hutchison, nicole.hutchison@stevens.edu; and **Hoong Yan See Tao,** hseetao@stevens.edu

ABSTRACT
In the US, the Department of Defense (DoD) is engaging in major digital transformation efforts to modernize the defense workforce, particularly the engineering acquisition workforce to ensure that the workforce remains technically competent. This effort was kicked off with the publication of its *Digital Engineering Strategy* in 2018. Digital engineering is necessary to update and support systems engineering practices to maintain a competitive advantage in data analytics, data science, computation technology, modeling, and simulation. The DoD envisions successful digital engineering transformation through the modernization of the way the Department designs, develops, delivers, operates, maintains, and sustains systems. One of the key aspects of this transformation is developing a digital engineering competency model to develop and sustain the workforce. In 2019, the DoD tasked the Systems Engineering Research Center (SERC) with the creation of a "Digital Engineering Competency Framework (DECF)" as part of its critical digital transformation and workforce development modernization efforts. Though created in the context of the US DoD, the DECF is not intended to be a defense- or US-centric and instead provides a foundation for any individual or organization looking to embark on digital transformation. This paper provides an overview of the DECF and its potential relationships with other competency frameworks.

INTRODUCTION

In early 2022, Ms. Stephanie Possehl, former Acting Deputy Director for Engineering Office of the Under Secretary for Defense (Research and Engineering) (OUSD(R&E)), wrote about the Engineering and Technical Management (ETM) vision for the US DoD's future. This vision included taking disparate but related career fields – Engineering (ENG), which includes systems engineers; Production, Quality, and Manufacturing (PQM); and Science and Technology Management (S&TM) – and replacing them with a new ETM functional area. This workforce encompasses the largest acquisition body in the DoD. With more than 72,000 members, it is nearly 40% of the defense acquisition workforce (Possehl 2022).

WHAT IS DIGITAL ENGINEERING?

DAU defines DE as "an integrated digital approach that uses authoritative sources of systems' data and models as a continuum across disciplines to support lifecycle activities from concept through disposal." In plain language, this means that digital engineering uses authoritative data and models to support the design, development, operation, maintenance, and retirement of systems. The strategic vision is to "securely and safely connect people, processes, data, and capabilities across a digital enterprise … using technologies such as advanced computing, big data analytics, artificial intelligence, autonomous systems, and robotics to improve the engineering practice." (DoD 2018) In the broader context, this also implies a transition to digital acquisition.

Though models and simulation are not new to the discipline of SE, DE represents a transformation from systems that are formally document-based to systems that rely on native models and data. "Digital transformation" is the term describing this transition from traditional, document-heavy, and event-driven acquisition and engineering approaches to a data- and model-driven approach that improves transparency and integration and allows improvements in existing processes. Full digital transformation requires both **digitization and digitalization**. Digitization is generally the easier of the two to tackle; moving from a physically printed document as the baseline to a PDF is a simple example of digitization. Digita-

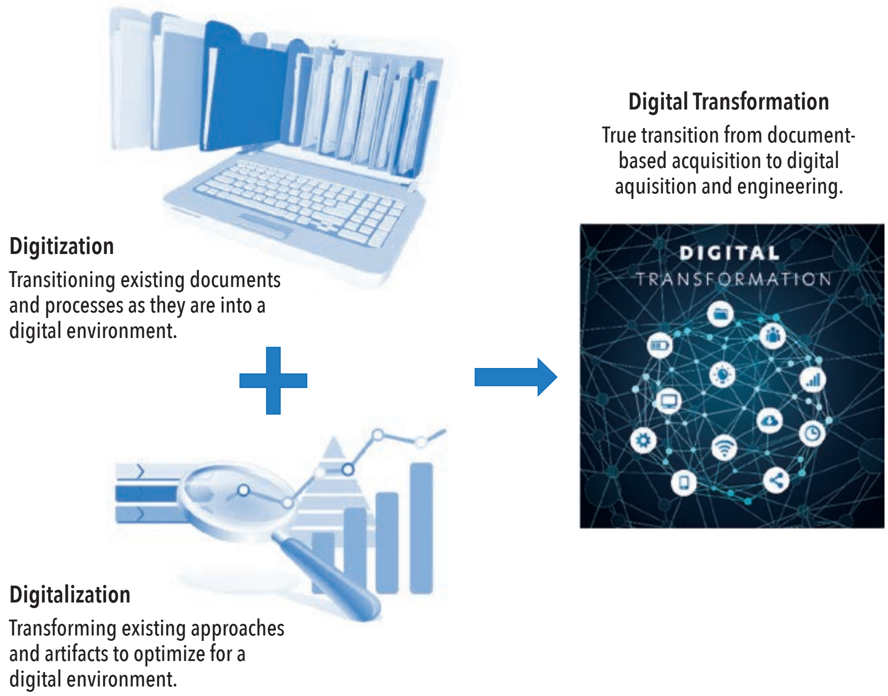


Figure 1. Digital transformation requires the combination of both digitization and digitalization. (Hutchison, See Tao, and Long – used with permission)

lization requires not that existing processes and artifacts be translated into an electronic environment but that these be reviewed to determine where they can and should be updated to improve effectiveness, efficiency, and transparency in a digital environment. It is only with thoughtful consideration of both aspects that any organization can achieve true digital transformation.

WHY DOES THE COMMUNITY NEED A NEW FRAMEWORK?

Beginning in 2020, the SERC team – comprised of researchers from the Stevens Institute of Technology, the Naval Postgraduate School, and the Georgia Tech Research Institute – worked with industry and government to understand critical skills for current and envisioned future DE skillsets. This included an analysis of existing competency frameworks such as:

- DAU Engineering Project Management competency models (DAU 2016a and 2016b)
- INCOSE *Systems Engineering Competency Framework* (INCOSE 2018)
- MITRE *Systems Engineering Competency Model* (MITRE 2007)
- NASA "Systems Engineering/Project Management Combined Competency Model" (NASA 2020)
- Helix *Atlas* Proficiency Model (Hutchison et al. 2020)
- IEEE *Software Engineering Competency Model* (IEEE 2014)
- US Department of Labor *Engineering Competency Model* (US Department of Labor 2017)
- SERC *Mission Engineering Competency Framework* (Vesonder et al. 2018)
- US Department of the Navy *Systems Engineering Career Competency Model* (SECCM) (Whitcomb et al. 2017)

The team started by developing a model of the existing competency frameworks, capturing each individual knowledge, skills, abilities, and behaviors (KSABs) statement from each model, and then reviewing to remove duplicates.

These competency frameworks provide insights into the SE, programmatic, general engineering, software, and modeling and simulation aspects of DE. A core takeaway, though, was that none of these were sufficient to cover the space. For example, Helix and the INCOSE framework discuss MBSE, but neither reflects how general SE skills would need to change to reflect an environment where digital approaches had become "business" as usual. This, therefore, became a core focus for the DECF work: updating the existing KSABs from these models to reflect work in a DE environment. The corollary to this was to add DE skills that were not covered in any of the existing competency frameworks.

THE LOGIC BEHIND THE DECF

The initial competency work resulted in over 1,000 KSABs. Even after clean-up, there are hundreds of individual skills required for DE. Note that no one individual is expected to have each of these KSABs, but a team performing DE must collectively have coverage across the breadth of the competencies. This number is too large for any individual to be able to grasp easily. The team, therefore, began exploring different mental models to explain the core facets of DE in a rationale way. These were reviewed with stakeholders in government, academia, and industry, and iterated over time. Figure 2 shows the final mental model for portioning the DE space in a way that is intended to be easily understandable. This provides the basis for the DECF structure:

- The foundation of DE is data. Understanding data storage, usage, sharing, security, and management are critical aspects of creating a DE process and environment that will provide a common backbone for DE work.

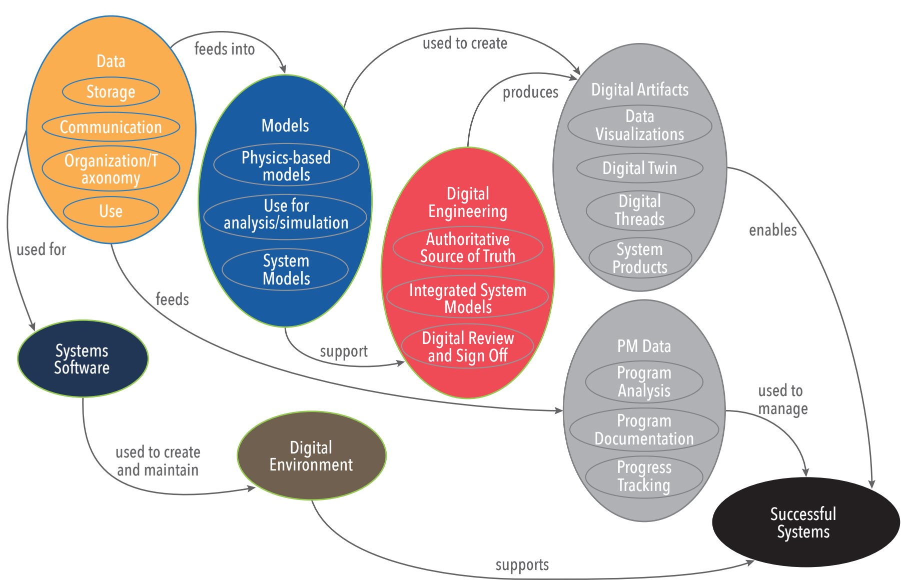


Figure 2. Logical model of the core aspects of digital engineering

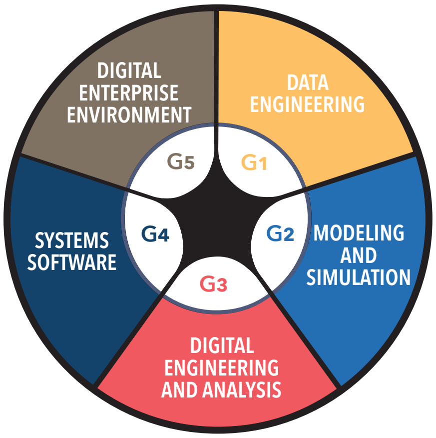


Figure 3. Digital engineering competency framework (DECF) v. 1.1 overview (SERC 2018)

- Data is the foundation for building models. Models in this sense mean abstractions of a system; simplifications that use data to create as realistic a representation of the system as necessary. Models are used for analysis and simulation, which create additional data and provide better insight into the system.
- Models provide the foundation for DE activities. Here, "DE" encompasses the systems engineering activities conducted in the digital environment as well as the acquisition-related activities.
- Digital engineering produces a variety of digital artifacts that are utilized by a variety of stakeholders and can be used throughout the system lifecycle. For example, a digital twin can be used in the development, but there are also examples of digital twins of operational systems that feed real-time data into the digital twin to allow predictive analysis. One example of this is the US Navy's Military Sealift Command, which uses data from ships in theater to better predict preventive maintenance needs (Systems Engineering Research Center 2022).
- In addition to the "engineering" context, DE should represent an integrated environment that uses authoritative data for all aspects of a program. Therefore, the PM aspects of a system should be integrated into the data flows.
- Data also feeds into the software aspects of a digital environment. There are two facets of systems software that are important in the DE context. The first are the principles of software engineering that will feed the processes and approaches used in a digital environment. The second is the software of the digital environment itself – the tools that support DE activities.

The vision is that if DE is implemented effectively, systems will be developed more effectively and efficiently, which will result in a higher percentage of successful systems.

Table 1. Individual Competencies in the DECF (Hutchison et al. 2021, used with permission)

G1 DATA ENGINEERING			
S1	Data Engineering	C1	Data Governance
		C2	Data Management
G2 MODELING AND SIMULATION			
S2	Modeling and Simulation	C3	Modeling
		C4	Simulation
		C5	Artificial Intelligence/Machine Learning
		C6	Data Visualization
		C7	Data Analystics
G3 MODELING AND SIMULATION			
S3	Digital Systems Engineering	C8	Digital Architecting
		C9	Digital Requirements Modeling
		C10	Digital Validation and Verification
		C11	Model-Based Systems Engineering Processes
S4	Engineering Management	C12	Digital Model-Based Reviews
		C13	Project and Program Management
		C14	Organizational Development
		C15	Digital Engineering Policy and Guidance
		C16	Configuration Management
G4 SYSTEMS SOFTWARE			
S5	Systems Software	C17	Software Construction
		C18	Software Engineering
G5 DIGITAL ENTERPRISE ENVIRONMENT			
S6	Digital Enterprise Environment Development	C19	Digital Environment Development
S7	Digital Enterprise Environment Management	C20	Management
		C21	Communications
		C22	Planning
S8	Digital Enterprise Environment Operations and Support	C23	Digital Environment Operations
		C24	Digital Environment Support
S9	Digital Enterprise Environment Security	C25	Digital Environment Security

The DECF v.1.1 has five main competency areas, which align with the discussion of the logic model of DE: data engineering; modeling and simulation; DE and analysis;

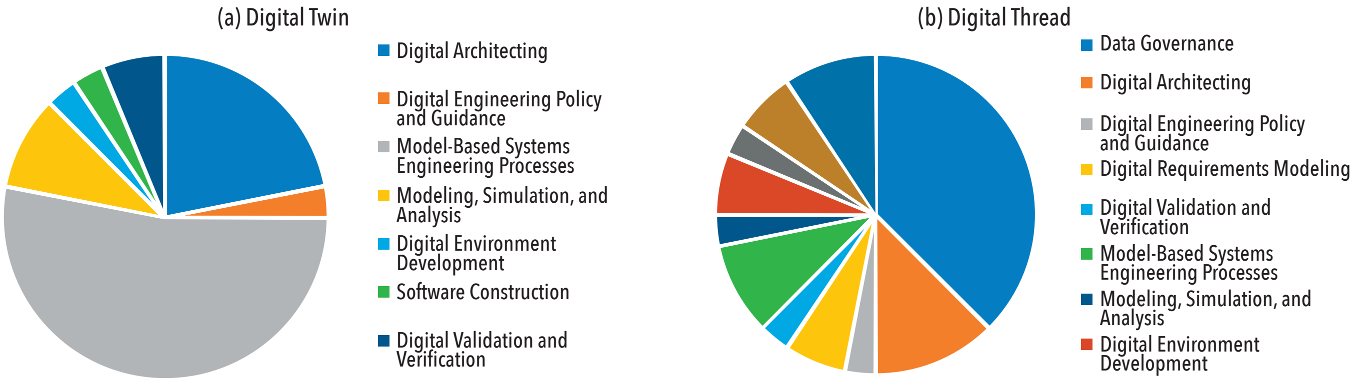


Figure 4. Distribution of digital twin- and digital thread-related KSABs throughout the DECF competencies.

systems software; and digital enterprise environment. (Hutchison et al. 2021)

Note that the full DECF can be downloaded from the SERC website, https://sercuarc.org/decf-review/.

The DECF v.1.1 contains six hundred fifty-nine (659) KSABs divided between these five competency areas. Each area contains individual competencies as shown in Table 1.

The KSABs within each competency are specifically targeted to digital engineering. Therefore, a broad competency area like "Communication" has a relatively low number of KSABs related to digital engineering, while a specific competency like "Model-Based Systems Engineering Processes" that is intrinsically linked with digital engineering has a higher number of KSABs.

The KSABs are distributed across five proficiency levels of the workforce: **Awareness**, **Basic**, **Intermediate**, **Advanced**, and **Expert**. At the Awareness and Basic levels, the KSABs represent wide-ranging fundamentals within a competency area. At the Advanced and Expert levels, the KSABs include the practice of detailed methods that make up the various applications of the competency area. The DECF is built to be a general framework that can be used to create specific competency models that are customized, so the KSABs cover the span of potential DE practices, even if all these practices are not utilized within every organization.

The competency list may be surprising in that expected items appear missing. Where are things like digital twins, threads, and artifacts? The simple answer is that these will integrate throughout the DE environment – and therefore, they are woven throughout the DECF instead of being stand-alone competencies themselves as shown for digital twins and digital threads in Figure 4. This illustrates that a number of competencies are required to implement or create these well.

Six additional foundational competency areas were identified as vital enablers for digital transformation and digital practices in general but are not necessarily "DE competencies". The researchers included them in the DECF but separated them from the five main competency areas. These competencies were deemed important to include in the DECF but also should be clearly separated from the five main DECF competency areas. The foundational competencies are identified as: digital literacy; DE value proposition; DoD policy/guidance; coaching and mentoring; decision making; and software literacy. Though this is a US DoD-focused set, these can easily be made more generic:

- General digital literacy (SFIA Foundation 2018): for most systems engineers, it is probably safe to assume a baseline understanding of operating in a digital world. However, when the aperture expands to include individuals across acquisition, this assumption may not hold true. General comfort with working digitally must be established for DE to be successful.
- DE value proposition: digital transformation is a long journey and will require a lot of work. If the workforce does not understand why they should go through this challenging process, it will require even more effort to make the transition. Regardless of the organization, making the value proposition for DE clear is a necessary step for embracing digital transformation.
- Policy and guidance: clear understanding of the organization's strategy and approach to DE. Without a clear vision for what this can be — hopefully linked to the value proposition — it will be an uphill battle for the workforce to make this transition.
- Software literacy: like general digital literacy, having a basic understanding of the major tools being used within the organization's DE context is a core competency on which the rest of DE work will be built.
- Decision-making and mentoring and coaching are general skillsets that are not specific to DE — however if these skills are not firmly established in the workforce, the gap will hinder the implementation of DE.

USING THE DECF

The DECF has contributed to several activities to support digital transformation in the DoD. The OUSD(R&E) has worked on creating a meta-model for engineering and technical management (ETM) competencies. Competencies from the DECF influenced the ETM model, particularly the inclusion of foundational digital literacy.

Partially, as a result of the DECF work, OSD has tasked the Defense Acquisition University (DAU) with developing a credential around DE. DAU is prioritizing its efforts to rapidly develop ETM credentials and other training materials to support this. Part of this effort includes DAU's partnership with SERC to develop a Simulation Training Environment for Digital Engineering (STEDE). Eventually, ETM certificates and credentials should have digital components that are consistent with the Department's goals for the ETM workforce. These goals are founded on competencies from the DECF, and the DAU approach is intended to provide a bridge between training and practical application of these skillsets. The STEDE is an ongoing research effort that continues to build upon the DECF. The results will be published on the SERC in Spring 2023. ■

REFERENCES

- DAU. 2016a. ENG: *Engineering Career Field Competency Model*. Version 2.0. Fort Belvoir, VA: Defense Acquisition University, Department of Defense.
- DAU. 2016b. PM: *Project Management Career Field Competency Model*. Fort Belvoir, VA: Defense Acquisition University, Department of Defense.
- Department of Defense. 2018. *Digital Engineering Strategy*. Washington, DC: US Department of Defense. June 2018.
- Hutchison, N., D. Verma, P. Burke, M. Clifford, R. Giffin, S. Luna, and M. Partacz. (2018). Atlas 1.1: An Update to the Theory of Effective Systems Engineers. Hoboken, NJ: Systems Engineering Research Center, Stevens Institute of Technology.
- IEEE. 2014. *Software Engineering Competency Model (v. 1.0)*. IEEE Computer Society. 2014
- INCOSE. 2018. *Systems Engineering Competency Framework*. San Diego, CA: International Council on Systems Engineering.
- MITRE. 2007. *Systems Engineering Competency Model*. McLean, VA: MITRE Corporation
- NASA. 2009. "NASA's Systems Engineering Competencies." Washington, DC: National Aeronautics and Space Administration. Retrieved May 31 2019 from: http://www.nasa.gov/pdf/303747main_Systems_Engineering_Competencies.pdf.
- NASA. 2020. "aPPEL Knowledge Services: PM&SE Career Development Framework." Washington, DC: National Aeronautics and Space Administration. Retrieved May 31 2019 from: https://appel.nasa.gov/career-resources/development-framework/.
- NAVAIR/NAVSEA Working Group. 2020 "Digital Competency Framework Overview: NAVAIR/NAVSEA Working Group." Presentation. March 2020.
- Possehl, S. 2022. "Engineering and Technical Management for Our Future." *Defense Acquisition Magazine*, 31 January. https://www.dau.edu/library/defense-atl/blog/Engineering-and-Technical-Management-for-our-future.
- SFIA Foundation. *Skills Framework for the Information Age 7: The complete reference. SFIA Foundation*. 2018.
- Vesonder, G., D. Verma, N. Hutchison, S. Luna, W. Miller, H.Y. See Tao, J. Wade. 2018. *RT-171: Mission Engineering Competencies Technical Report*. Hoboken, NJ: Systems Engineering Research Center (SERC), Stevens Institute of Technology. 30 April 2018. SERC-2018-TR-106.
- Whitcomb, C., R. Khan, C. White. 2017. *The Systems Engineering Career Competency Model, Version 1.0*. Monterey, CA: Naval Postgraduate School. 30 March 2017.

ACKNOWLEDGMENTS

There were extensive teams that conducted the foundational research on which this article is based. We would like to thank these researchers: Kara Pepe, Mark Blackburn, and Dinesh Verma, Stevens Institute of Technology; Cliff Whitcomb, Rabia Khan, Naval Postgraduate School; and Russell Peak and Adam Baker, Georgia Tech. We would also like to thank the government and industry collaborators who supported the DECF work, in particular Phil Zimmerman, who provided the initial vision and oversight on this effort.

ABOUT THE AUTHORS

Dr. Nicole Hutchison is a Principal Investigator (PI) and research engineer at the Systems Engineering Research Center (SERC). Her primary work through the SERC has been in human capital development research. This has included the development of competency frameworks for systems engineering (the Helix project), digital engineering, and mission engineering. Currently, Dr. Hutchison is the PI for the Simulation Training Environment for Digital Engineering (STEDE), a project that is developing realistic models to be used in training the DoD acquisition workforce in a way that builds hands-on digital engineering skills. She previously served on the BKCASE research team, which resulted in the development of the Guide to the Systems Engineering Body of Knowledge (SEBoK). She is currently the Managing Editor for the SEBoK and the Lead Editor for the "Enabling Systems Engineering" section of the SEBoK. Before joining the SERC, she spent 5 years working for Analytic Services, Inc., supporting the US Departments of Defense, Homeland Security, Health and Human Services, and Justice. She holds a PhD in systems engineering from Stevens and is a Certified Systems Engineering Professional (CSEP) through INCOSE.

Dr. Hoong Yan See Tao is a Research Project Manager for the Systems Engineering Research Center (SERC). She is also a researcher on a current SERC research task titled "DAU Digital Engineering Simulations" and the Co-PI of the Defense Data Grand Prix. Previously, Yan was a SERC Research Associate and Postdoctoral Fellow where she conducted research in mission engineering, digital engineering, and performed project management roles with several human capital development and systems engineering and systems management transformation projects. Yan graduated with a Ph.D. in Systems Engineering at Stevens Institute of Technology in 2017. Yan was the President of the INCOSE Student Division at Stevens in 2013. Yan received her B.S. and M.S. in Electrical Engineering from the University of Texas at El Paso (UTEP). At UTEP, she worked in the Research Institute for Manufacturing and Engineering Systems and was the team leader for two NASA and Texas Space Grant Consortium design challenges.

For the Journey to Expertise in Systems Engineering, Enhance the Path with Shu Ha Ri

Fred Y. Robinson, frobinson@mitre.org

■ **ABSTRACT**
This paper offers a perspective for considering enhancements to the current programs for developing systems engineering professionals, incorporating consideration for developing characteristics of expertise and mastery throughout. Shu Ha Ri represents an approach for three phases of mastery development, established in ancient practices such as some martial arts and mimicked in many current approaches: beginner > intermediate > advanced, apprentice > journeyman > master, bachelors > masters > doctorate.

Anders Eriksson's research on expertise depicts three levels of progression, naïve practice > purposeful practice > deliberate practice, building on Bloom's earlier three phases of development. However, Eriksson's model is limited to domains where the demonstration of expertise can be characterized, is well understood, and is measurable or at least objectively evaluable by existing domain experts.

Most systems engineering expert achievement certifications that exist are still relatively subjective or IKIWISI (I'll know it when I see it) evaluations by the subject's future peer experts. For a given population of experts in systems engineering, there is a shared thematic set of highly diverse experiential assessment characteristics which diverge in orthogonal dimensions from some of the earlier assessment levels.

David Epstein suggests that the power of generalists comes to play more when experts address wicked problems (those lacking a pre-ordained approach for solving) than when specialists address kind problems (the opposite). Kind problems are not necessarily easy to solve, but the route is well defined. Solving wicked problems without exemplar solutions often requires the generalist's leveraging of analogic thinking, and the recognition and possible synthesis of matchable patterns (isomorphisms) learned from diverse experience sampling of other domains, not merely relying on T-shaped or Pi-shaped knowledge.

Using the Shu Ha Ri framing presents an opportunity to consider enhancements to earlier systems engineering practitioner development stages towards excelling beyond emergence and effectiveness.

■ **KEYWORDS:** Shu Ha Ri; specialists; generalists; systems engineering; expertise; analogic thinking; wicked problems

INTRODUCTION

In a 2019 NDIA presentation (Robinson 2019b), I posed a query asking, "Where is systems engineering's ***dark matter***?" Many efforts to date have approached identifying the explicit characteristics of our practice, or the **explicit knowledge**, often documented through language or other representations via artifacts or models. Explicit knowledge is discernable, much like the observable celestial objects in the universe. Disciplinary technical specialties and methods provide galaxy-like context to coalesce systems engineering's explicit knowledge in collections of capabilities or competencies. In contrast, ***tacit* knowledge** (thoughts, understanding) refers to *that which you know but cannot say*, such as perceptions, beliefs, and feelings (Polanyi 1966). When considering what constitutes an expert in systems engineering, mimicking traditional mastery progression paths that reflect honing well-specified talents to perfection can fall short. Systems engineering experts tend to share certain common attributes or characteristics, yet they are mostly identified through IKIWISI, or I'll know it when I see it (Boehm 2000) peer assessments by prior recognized disciplinary experts, somehow defying or detouring from the explicit linear characterizations found in the earlier levels of mastery. The question remains how to define and link the tacit attributes common across systems engineering's practitioners to the explicit paths they experienced to attain expert status, which I would call the dark matter of systems engineering.

How do we go about identifying that which we cannot easily see yet we believe is necessary to complete our understanding? Exploring a mastery approach from Japanese culture as popularized in martial arts seemed intriguing, though I have no first-hand experience. Japanese cultural disciplines have long histories of defining

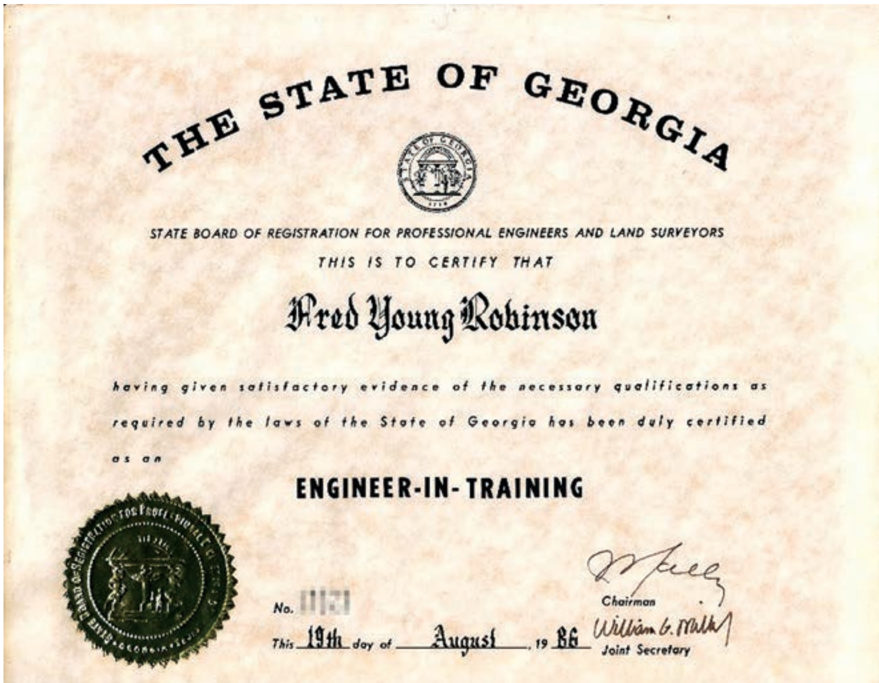

THE STATE OF GEORGIA
STATE BOARD OF REGISTRATION FOR PROFESSIONAL ENGINEERS AND LAND SURVEYORS
THIS IS TO CERTIFY THAT
Fred Young Robinson
having given satisfactory evidence of the necessary qualifications as required by the laws of the State of Georgia has been duly certified as an
ENGINEER-IN-TRAINING
No.
Chairman
This 19th day of August, 19 86
William G. Malk
Joint Secretary

Figure 1. Engineer in training certificate

and developing disciplined mastery. If the goal is to master martial arts training, such as Aikido, it should not be pursued casually. True mastery requires students to be committed for the long haul, as the concept of eventually becoming a master appears inherent at all levels of training. A dedicated student would be an anomaly to endeavor towards becoming the *best beginner* or an *impressive intermediate* performer, aiming only to stagnate somewhere in the middle of what should be their journey to mastery. Yet, the journey to expertise in systems engineering can be seen as incomplete, depending how the characterization of an "expert" in systems engineering would compare to a master practitioner of Aikido. Do the achievements that we use to define progression through the early stages of systems engineering mastery resonate with the attributes that describe our disciplinary experts as would occur in martial arts, or does the picture of an expert only form coincidentally adjacent to these achievements?

SHU HA RI FOR SYSTEMS ENGINEERING?

Shu Ha Ri represents three-phases of mastery pursuit, rooted in a triplet of ancient Japanese constructs (Dease 2010; Koch 2020; Yuta 2021) simplified as: **Shu** to learn the basics ("follow" – to *know*), **Ha** to learn tools ("seek" – to *do*), and **Ri** to extend beyond just using the tools and demonstrating skills ("leave" – to *excel*). Researchers suggest this triplet originated with a 15th-century sword master's teachings, though it has persisted in modern times particularly for the martial art Aikido as well as through influences to other diverse knowledge traditions including flower arranging, theater, sculpture, and weaving (Chung and Tan 2017; Takamura 2004).

The Eastern worldview on developing expertise encourages continual entry to learning new knowledge that can progress to greater levels of mastery in independent and parallel experiences, where one's journey is a composite of overlain Shu Ha Ri experiences. Alternatively, our Western worldview norm is to chunk knowledge or skills into discrete levels or stages, where crossing a threshold implies being done with the prior level and not having to return to that stage, where the journey is to always move upward (Rother 2016). Westerners' interpretation of Shu Ha Ri to be a linear, chunked, and stepwise journey constrains one's potential for flexibility or agility of continual learning. As I was receiving my first degree, I also took and passed the test precursor to a Professional Engineer (P.E.) certification, to become an "Engineer in Training" as depicted in Figure 1. While I never pursued the P.E., I continue to see myself as an *engineer in training*, seeking to continually learn more. Since the Eastern view of gaining knowledge in a particular Shu, Ha, or Ri phase is not independently holistic, we should continue to learn basics in new areas concurrently as we hone skills or evolve new techniques in others.

The popular approach for development towards mastery in our systems engineering community is primarily based upon a basic three-stage progression, first decomposing our discipline into sets of foundational basics for us to learn (Shu), and then definitions of sets of competencies (skills, tools, and methodologies) to be individually mastered (Ha). Achieving a composite of singular masteries across the full suite of individual systems engineering competencies might not cleanly translate to being an expert systems engineer. One grounded theory study seeking to characterize our discipline's experts resulted in over a dozen themes of demonstrated attributes, capabilities, and behaviors (Armstrong and Wade 2015), many of which were in orthogonally different dimensions from the sets of reduced skills or competencies at the lower levels (detailed later). The orthogonal shift reflects the expected departure of a Ri stage. Many of these cross-cutting themes might appear during one's journey to mastery, perhaps through serendipity instead of pre-meditated guidance or self-development. By examining our peers' approaches for their journey to overall systems engineering expertise, I hope to identify

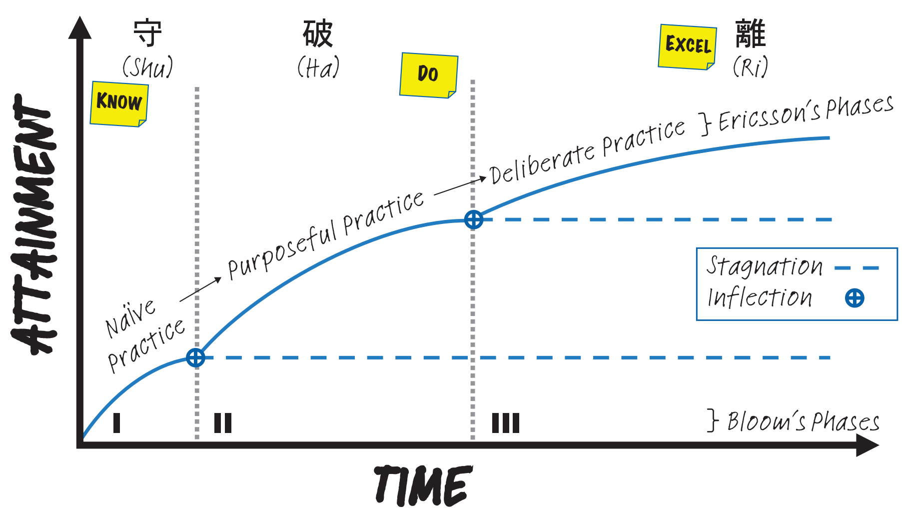


Figure 2. Ericsson, Bloom, and Shu Ha Ri

potential reframing opportunities amongst the multitude of apparent strengths of our current approach, leveraging the framing of Shu Ha Ri.

SHU HA RI FOR SPECIALISTS

Anders Ericsson's research on expertise (Ericsson and Chamess 1994) which he expanded upon in his book **Peak: Secrets from the New Science of Expertise** (2017) also depicts three levels of progression akin to **Shu Ha Ri**, which he refers to as *naïve practice → purposeful practice → deliberate practice*, building on Bloom's earlier definition for three phases of skill development (Bloom 1985). Eriksson's model is limited to domains where the demonstration of expertise can be characterized, is well understood, and is measurably honed or at least objectively evaluable by existing domain experts. I have attempted to depict in Figure 2 a map that begins to overlay these frames for attainment towards expertise. Each practice phase includes incremental learning that typically plateaus, requiring an inflection point to break into the next style of practice and level of learning. Without such an inflection point, stagnation results, where one could easily endure years of repetitive execution yet not actually learn anything new. Settling into these stagnation conditions short of expertise is one of the ways Ericsson calls out Gladwell for *missing the point* by focusing on the 10,000 hour duration element of attaining expertise in the book **Outliers** (2008).

Three-part mastery development exists in many current developmental journey triplets: beginner → intermediate → advanced, apprentice → journeyman → master, bachelors → masters → doctorate. My alliterative triplet, what I call "the three Es," has been: emergent → effective → expert.

There are multiple approaches across our discipline for training at the entry levels for the ***emergent*** systems engineering practitioner which also align to the *naïve practice* construct and the ***Shu*** "obey" or "follow" stage. Naïve practice consists of learning the vernacular, the rules, and the basic execution necessary to be a participant. In tennis, for example, learning the basics of holding a racquet, hitting the ball, and the rules of playing a game allows one to independently play the sport with others. Feedback on learning in naïve practice is often limited to validations of knowledge and simple confirmations for the execution of basic defined behaviors. INCOSE (International Council of Systems Engineering) offers the Systems Engineering Professional (SEP) program, where the "certified" level of accomplishment is based primarily upon a test of basic knowledge from the INCOSE Systems Engineering Handbook (2015), along with a modicum of demonstrated performance in the field. It defines similar programs for other organizations like the "foundational" level of Engineering & Technical Management at Defense Acquisition University (DAU) or less formal indoctrination programs at other agencies or corporations.

The characteristics of an ***effective*** systems engineer as might be achieved through *purposeful practice* and characterized by the ***Ha*** "detach" or "digress" stage is to go beyond the basics established in Shu. Purposeful practice means establishing more formal relationships with trainers, coaches, or mentors. These journey guides are not only already accomplished practitioners but can encourage their learners to break from one's comfort zone while also providing constructive feedback toward incremental improvement.

The Atlas research effort (SERC 2018a) by the Systems Engineering Research Center (SERC) was a grounded theory study to determine the attributes of effective systems engineers. Atlas facilitated the creation of the Helix model (SERC 2018b) which decomposes and measures the discovered attributes along six thematic axes. Other efforts in our discipline have addressed establishing systems engineering competency models. These models decompose sets of the discipline's performance activities, like INCOSE's framework model with its competencies broken into five groups and 36 core areas, measured across five proficiency levels up through *expert* (Gelosh et al. 2018). Such a model begs the question of where to find emergence for holistic expertise in systems engineers. Does the whole of being an expert merely occur as the *sum of the parts* from being assessed to the individual expert levels of the competency framework? Do the behaviors and attributes of those we call expert systems engineers emerge through other behaviors like synthesis, systemic thinking, critical thinking, allegorical thinking, or other dimensions that can cut across a competency model's threads? How does a highly effective systems engineer engender an inflection point to go beyond the amassing of effectiveness scores for competencies or capabilities characterized by one's discipline?

The ***Ri*** stage is where one would "leave" or "separate" from the foundations and transcend into being an *expert* systems engineer. Most ***expert*** achievement certifications that exist for systems engineering are still relatively subjective or IKIWISI (I'll know it when I see it) evaluations by future peer experts. For a given population of experts in systems engineering, there are typically only a thematic set of highly diverse experiential assessment characteristics (Armstrong and Wade 2015). These characteristics diverge in mostly orthogonal directions that do not align to the foundational constructs of Shu and Ha, nor is there any clear emergence path. INCOSE does offer an Expert SEP (ESEP) assessment level, characterized by creating a large document detailing experiences across a 25-35+ year career in the field (depending upon associated degrees), but more so requires other existing ESEP members to vouch for the applicant. A candidate with a 20-30+ year career (depending upon associated degrees) can also apply along with having passed the SEP fundamentals test, while still needing the existing ESEPs to vouch for their candidacy. The INCOSE ESEP approach provides a structure for assessing durations and depths of experiences, seeking the recounting of 2+ years in at least 6 of 14 functional areas, with added peer validation (INCOSE 2017b; 2017a). However, these assessments applied through the middle phase of career / expertise progression towards the achievement of ESEP recognition only address a portion of the characteristics uncovered by Armstrong & Wade's study.

In the triplet for Ericsson's "Peak" model, he constrains the applicability of his definition of expertise evolution, or attainment of mastery, to fields that could be best associated with specialists. An aphorism I learned early in my career was that *specialists learn more and more about less and less until they know everything about nothing* (Robinson 2019a). Deliberate practice is honing one's performance to definitions that characterize only the top performers, the extreme specialists, under intense training under other discipline experts, peers who are uniquely qualified to provide the necessary constructive feedback. Systems Engineering expertise does not lend itself to a focused honing of a discrete set of highly refined and difficult to accomplish behaviors found in highly specialized fields. However, becoming an expert in playing violin, driving a taxi in London, performing delicate surgeries, competing for gold in Olympic sports, these all have very narrow definitions for top performance and achieving the ultimate levels of success. Tradecraft masters such as electricians, plumbers, and builders also address their toughest challenges by relying on well-established and narrowly defined practices and techniques. Employing divergent thinking or applying elements of other disciplines at this level of performance could have disastrous results.

Figure 3 depicts how the various mastery triplets discussed previously align to each other, and in the context of the stages for Shu Ha Ri. The similarities can break down

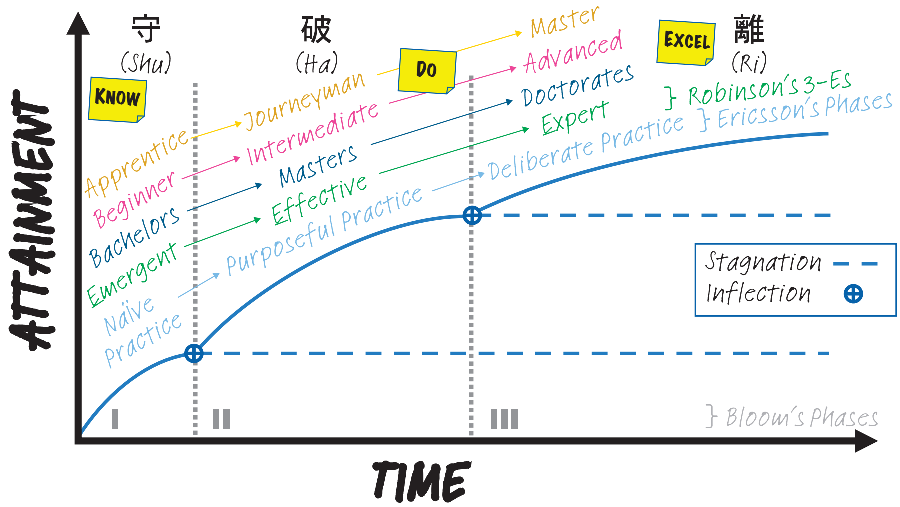


Figure 3. Additional parallel triplets in alignment to Ericsson, Bloom, and Shu Ha Ri

in the Ri stage however depending upon the nature of the discipline in which mastery is pursued, particularly where such expertise might not fit Ericsson's assertion for a honed and tightly measurable *peak* attainment.

SHU HA RI FOR GENERALISTS

A corresponding yet opposing aphorism (to the specialist aphorism above) is that *generalists learn less and less about more and more until they know nothing about everything* (Robinson 2019a). While considering some other disciplines' top achievers (systems engineering), a different set of circumstances characterize the execution and recognition of mastery. David Epstein's book **Range: Why Generalists Triumph in a Specialized World** (Epstein 2021) offers a perspective that might help explain the divergence of dimensions of systems engineering expertise in the ***Ri*** stage. Epstein suggests that the power of *generalists* comes to play when experts address **wicked problems** (those lacking a pre-ordained approach for solving) than where specialists address kind problems (the opposite). **Kind problems** alternatively are not necessarily easy to solve but the routes to solutions are well defined (such as in Ericsson's Peak model). Solving wicked problems in the absence of exemplar solutions (as kind problems have) often required the generalist's leveraging of systemic thinking habits and allegorical thinking through recognition and synthesis of matchable patterns (isomorphisms) learned from diverse experience sampling of other domains.

Another way to describe wicked problems is captured by Rosalind Armson in **Growing Wings on the Way** (2011) as **messy situations**, or those ambiguous and fuzzy problems for which solutions need to be discovered, those that are not just difficult but achievable. Leveraging systems thinking, or more aptly described as learning to think systemically, she suggests multiple habits of a generalist nature to facilitate discovery, such as varying amongst reductionist, holistic, scientific/mechanistic, socio/mechanistic, and constructive thinking, as well as ways for escaping traps or blocks that may come from others' perceptions or preconceptions. She posits that a messy situation may not have particularly right or wrong solutions, perceptions of which are often distinct to the individual observers or solvers. Learning which approaches may deliver preferred outcomes is important, achieved through different approaches to explore what makes each problem messy. The journey however can present fallacies which Armson characterizes particularly well concerning diagramming efforts that might be associated with exploring messy problems. The two most powerful exposed fallacies for me were, 1) that no one diagrammed representation is more real, even though creators often assume theirs is not only the right one but that it will evoke shared understanding for all others (which doesn't necessarily occur), 2) that a finished representation must inevitably become intuitively obvious having benefited from all the rounds of drafting or development, whereas it's only the creator's hindsight that engenders this perception, which may not be shared by others who weren't on the journey of discovery.

Success for Epstein's generalists comes from developing/leveraging their strengths in abstraction from what they know for use with the unknown, through the application of analogical thinking, especially with the synthesis of multiple disparate domains of knowledge. Analogic thinking is akin to **adaptive thinking** identified in the US Army's Think like a Commander approach (Lussier and Shadrick 2003). This approach recognizes that officers required focused training to effectively deal with unanticipated events or circumstances while leading military exercises during what would be chaotic circumstances in the heat of battle. This approach expected commanders to operate adaptively under the stress of a multitude of conflicting demands for their attention and cognitive processing, and therefore training environments should reflect the reality of (wicked) chaos versus repetitive isolated honing of selected skills in calm, well defined, and reduced complexity situations. Expert adaptive thinking requires extensive training across complex conditions to develop the habits that allow for assuring success or at least minimizing harm.

The power of the generalist in Epstein's Range is more than just having T-shaped or Pi-shaped ("π") knowledge, known as having depth in one (T) or two (Pi) vertical capability strength areas with a shallower

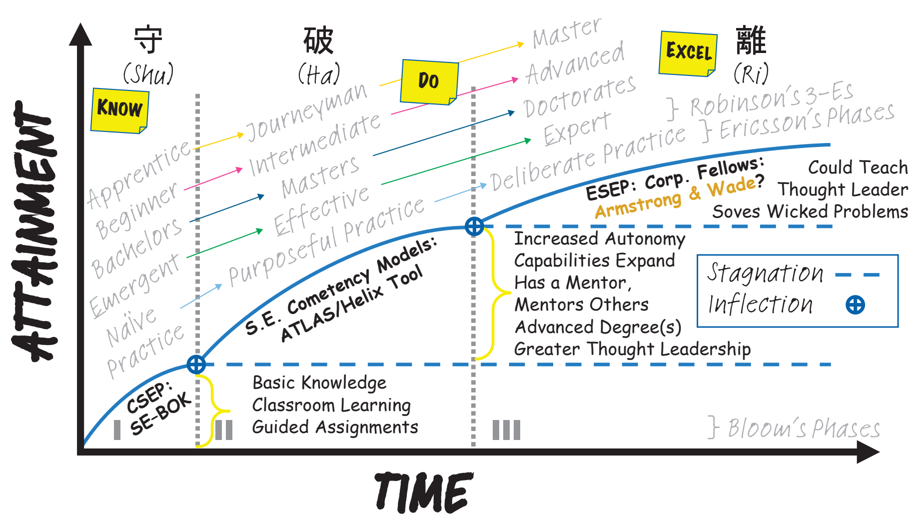


Figure 4. Mapping systems engineering assessment frames to the triplet levels

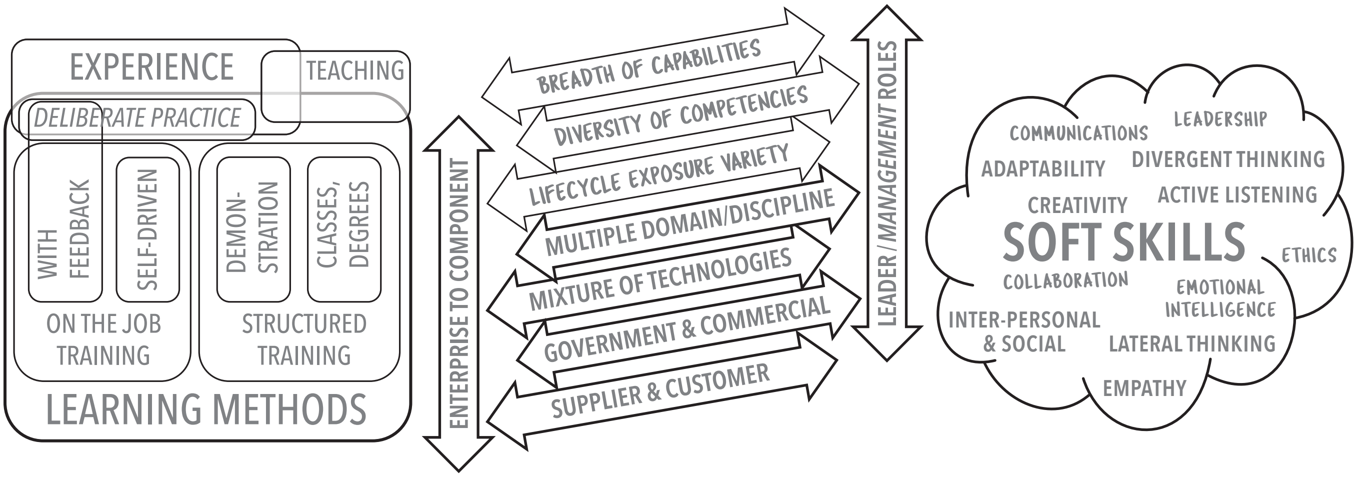


Figure 5. Learning methods, practice extents, and soft skills depiction of systems engineering expertise

breadth across multiple complementary disciplines or fields (Trogstad, Kokkula, and van der Aker 2021). Some also might mistakenly limit or dismiss the power of the generalist, such as by using the reference as a dash-shaped ("—") professional, diminishing one who is broadly knowledgeable in many disciplines (a *jack of all trades*) as being a *master of none* (Demirkan and Spohrer 2015). Generalists' ability to wade through the ambiguity or novel complexity of wicked problems comes from developing foundational experiential breadth, and this only occurs through significant disciplinary sampling across a multitude of domains, according to Epstein.

A generalist in Epstein's frame is more akin to a **polymath** who can demonstrate or achieve high mastery levels across broad sets of disciplines. In what is apparently a modern twist on the time-worn aphorism *being a jack of all trades and a master of none*, some suggest that the full thought ends with, is *ofttimes better than a master of one* (Epstein 2020). So, the question is, how can Shu Ha Ri inform the path to systems engineering expertise?

EXPERTISE OUT OF SHU HA RI

Armstrong & Wade (2015) suggest there is a dearth of research on the specifics of how systems engineers reach their top level of expertise, although how long it takes is somewhat understood, referring to departing experts *spending years in the trenches*, or more so multiple decades. Interviewed experts tended to focus on what path their journey took versus how they developed their expert qualifications.

Figure 4 extends the Figure 3 triplets view with references under the curves within each phase, selected existing assessment frameworks for systems engineering are listed which inform and shape development within the related stage. Also depicted are more generalized typified characteristics of each level/phase as personal attainment of systems engineering mastery grows.

The Armstrong & Wade (2015) study also suggests a dearth of research regarding the alignment of systems engineering expertise to Ericsson's and Bloom's observations regarding the triplets of mastery stages. Naïve practice or the Shu (*follow/know*) stage can logically be correlated to the learning of the basics of systems engineering, whether absorbed from memorizing the Systems Engineering Body of Knowledge (SE-BOK) or studying to pass the Systems Engineering Professional (SEP) knowledge exam based upon the INCOSE Systems Engineering Handbook. The basics covered in these examples have been deconstructed and reduced by previous generations of systems engineers by using their foundational experiences. Purposeful Practice or the Ha (seek/do) stage can also logically be correlated to expanding one's disciplinary knowledge through learning and mastering the tools and methods that previous and current generations of systems engineers established. Yet, as just having a toolset does not make one a carpenter, there are a variety of models that assess the capability of using tools and methods to do systems engineering tasks, including competency frameworks such as offered by INCOSE, or the Atlas/Helix assessment. Elements within these assessments do address some isolated aspects of synthesis, integrative approaches, or thinking systemically, yet primarily they exist as a decomposed reductive set of constructs.

Understanding the basics of systems engineering (Shu) satisfies the knowledge dependencies for understanding how to use tools and apply the methods for effective practitioner performance (Ha). Similar to the toolbox to carpenter analogy, however, becoming an "expert" for each competency correlates with a few of the multiple thematic extractions of characteristics of expert systems engineers uncovered by Armstrong & Wade (2015). I offer a notional depiction of the groupings of the authors' three primary frames in Figure 5, where many of these sub-elements are typically found toward the end of one's mastery journey and reflect a departure from just being able to muster the capabilities, very much like the expected expand and leave indicative of the Ri stage.

The learning methods to the left more so characterize "how" one can achieve knowledge across earlier stages. The propensity to teach is novel if non-existent when reviewing the competency framework and should not be conflated with defining practice areas or being the "go-to" practitioner. Ericsson's descriptions for purposeful practice and deliberate practice emphasize learning is most effective with focused external feedback, often lacking in structured training and self-driven on-the-job training. The professional practice extents as represented by double-ended arrows define a significant number of additional and novel capacity dimensions of generalists other than the extent of capabilities, competencies, lifecycles, and management elements indicated by the stylized font. Soft skills were a major thematic observation for the authors but not fully detailed in the discussion of their findings. While the competency framework addresses some common soft skills (in the stylized font) mostly through the generic "professional" subset, many additional soft skills are typified by the practices' experts.

SHU HA RI IN ACADEMIA

Attaining expertise in systems engineering does seem to have a Ri-like parallel in the academia triplet. Bachelor's degree recipients have established basic

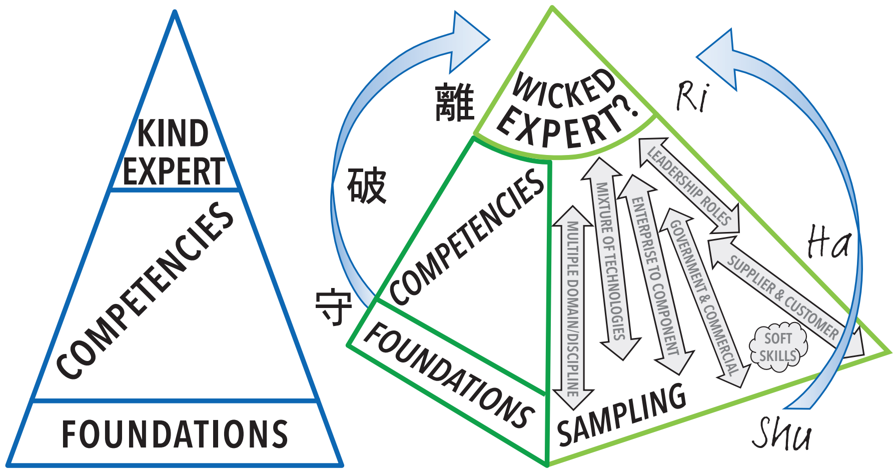


Figure 6. Kind expert versus wicked expert developmental planning

knowledge of the academic discipline. A master's degree demonstrates one's deeper exploration of the known skills and methods of the academic discipline. Yet, one's doctoral pursuit requires a departure from that which is known, while still requiring the foundations as a predecessor, to create something novel to add to the body of knowledge in the field. Doctoral achievements reflect the solving of some previously unsolved (wicked) problem. Similar to the challenges faced by effective systems engineers who find themselves with no journey definition available on how to become experts in our discipline, graduate students are often equally unprepared to shift into their unique type of deliberate practice demanded by doctoral pursuits. As evidenced by the high levels of attrition from doctoral pursuits, many academics possibly pride themselves on creating hurdles toward achieving doctoral expertise, such as excessively rigid application acceptance criteria or sustaining discouraging environmental conditions that engender departure, whether it's intentional or coincidental (Ali and Kohun 2006). The attributes that are contributing factors to ABD (all but dissertation) discontinued doctoral pursuits are often independent of the candidates' successes in their earlier degree pursuits. Ali and Kohun focus on how to make doctoral programs more responsive to students' needs in an environment that is fundamentally different from any prior experiences for which they have little influence to effect change.

Given the smaller scale of systems engineering mastery development in comparison to academia, we can shift towards enhancing the earlier mastery journey phases to better enable future experts' capacity for handling messy situations and solving wicked problems. More so, both the systems engineer's and the academic's transition across naïve practice and purposeful practice stages currently lack elements specifically to prepare one for handling wicked problems, unless serendipity has intervened. Ironically, attaining a doctorate in systems engineering demonstrates becoming an extreme specialist on one's research topic, whereas attaining expertise in the practice of systems engineering seems better aligned to being an extreme generalist.

WHAT CAN WE BE THROUGH SHU HA RI?

As suggested above, academia's approach would be to create a more supportive environment while attempting to mediate or mitigate as many addressable barriers as possible within their third mastery stage. This is necessary given the immense diversity across academia, the lack of a sufficient centralized notion of governance, as well as the current culture's belief that one's mastery path is complete through attainment (and acceptable stagnation) of a masters' degree. Thankfully, the discipline of systems engineering has the benefit of stronger collective governance such as through INCOSE and SERC, as well as a culture that encourages the potential achievement of expertise for all practitioners.

Whether laid end-to-end or stacked bottom-up-to-top, the systems engineers' progression through the emergent stage and then into and through to becoming effective practitioners starts as a sequential construction on a strong foundation. Yet, the expert practitioners who address wicked problems present themselves as having characteristics that emerge in a departure from, yet built upon, the currently defined foundations of the earlier stages, rather than deliberate honing precision solution performances found in disciplines more focused on kind problems. Figure 6 depicts one way to contrast potential differences in developmental planning for attaining kind expertise versus wicked expertise.

Planning the development of practitioners via focusing on a reduced-out set of foundational knowledge and competencies suits the purposes of creating expertise for disciplines that address solving kind problems. However, acknowledging the commonality across the developmental stages of systems engineers who later achieve expertise through demonstrating their abilities in solving wicked problems, we should be seeking ways to integrate greater sampling in concert with the attainment of foundational knowledge and competency mastery. Learning new aspects of applying one's refined systems engineering competencies across the extents of any of these alternate dimension elements is the very nature of having a Shu experience concurrently while potentially developing in Ha or Ri aspects of knowledge or skills. Leveraging Shu Ha Ri does not suggest there is a single three step journey to overall mastery, as westerners perceive it, but rather that mastery can be attained through the integration of multiple and asynchronous phased developmental triplet experiences (the Eastern worldview).

Inherent in the phased constructs of Shu Ha Ri is that one's journey of steps should lead towards expressions of expertise that separates from and expands the discipline, not just to provide for sharply honed precision for dealing with well-understood specialized challenges. One cannot develop an analogical thinking aptitude after the fact. They must experience the dark matter for expertise in systems engineering tacitly through the sampling of the extents we find later in our experts, as part of the plan for the journey. This way, we do not rely solely on serendipity for our disciplines' experts to emerge from within the community of practice. ■

Approved for Public Release; Distribution Unlimited. Public Release Case Number 22-0821

The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions, or viewpoints expressed by the author.

REFERENCES

- Ali, A., and F. Kohun. 2006. "Dealing with Isolation Feelings in IS Doctoral Programs." *International Journal of Doctoral Studies* 1: 021-033. https://doi.org/10.28945/58
- Armson, R. 2011. *Growing Wings on the Way: Systems Thinking for Messy Situations*. Axminster, UK: Triarchy Press.
- Armstrong, J. R., and J. Wade. 2015. "Development of Systems Engineering Expertise." *PROCS Procedia Computer Science* 44: 689–98. https://doi.org/10.1016/j.procs.2015.03.026
- Bloom, B. S. 1985. "Generalizations about Talent Development." In *Developing Talent in Young People*, edited by B. S. Bloom, 507–49. New York, US-NY: Ballantine Books.
- Boehm, B. 2000. "Requirements That Handle IKIWISI, COTS, and Rapid Change." *Computer* 33 (7): 99–102. https://doi.org/10.1109/2.869384
- Chung, L., and K. Tan. 2017. "The Unique Chinese Innovation Pathways: Lessons from Chinese Small and Medium Sized Manufacturing Firms." *International Journal of Production Economics* 190 (August): 80–87. https://doi.org/10.1016/j.ijpe.2016.09.004
- Dease, D. 2010. "The Meaning of Shu Ha Ri." *Dojo Blog*, 29 June. http://www.kimusubiaikido.com/blog/meaning-shu-ha-ri.
- Demirkan, H., and J. Spohrer. 2015. "T-Shaped Innovators: Identifying the Right Talent to Support Service Innovation." *Research-Technology Management* 58 (5): 12–15. https://doi.org/10.5437/08956308X5805007
- Epstein, D. 2020. "How Falling behind Can Get You Ahead." Filmed April 2020 at TEDxManchester, Manchester, UK. Video, 14:25. https://www.youtube.com/watch?v=BQ2_BwqcFsc.
- ———. 2021. *Range: Why Generalists Triumph in a Specialized World*. New York, US-NY: Riverhead Books.
- Ericsson, K. A. 2017. *Peak: Secrets from the New Science of Expertise*. New York, US-NY: Houghton Mifflin Harcourt.
- Ericsson, K. A., and N. Chamess. 1994. "Expertise: Its Structure and Acquisition." *American Psychologist* 49 (8): 725–47. https://psycnet.apa.org/doi/10.1037/0003-066X.49.8.725
- Gelosh, D. S., M. Heisey, K. Nidiffer, J. Snoderly, and R. Beasley. 2018. "Version 1.0 of the New INCOSE Competency Framework." *INCOSE International Symposium* 28 (1): 1778–86. https://doi.org/10.1002/j.2334-5837.2018.00583.x
- Gladwell, M.. 2008. *Outliers: The Story of Success*. New York, US-NY: Back Bay Books.
- SERC (Systems Engineering Research Center) Technical Report SERC-2018-TR-101-A. 2018a. Atlas 1.1: An Update to the Theory of Effective Systems Engineers. Hoboken, US-NJ: Stevens Institute of Technology.
- SERC (Systems Engineering Research Center) Technical Report SERC-2018-RT-173. 2018b. Helix Technical Report. Hoboken, US-NJ: Stevens Institute of Technology.
- INCOSE. 2015. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. Version 4. San Diego, US-CA: INCOSE.
- ———. 2017a. "ESEP Instructions for Completing Form 41." https://www.incose.org/docs/default-source/certification/form-42-esep-instructions-for-completing-form-41.pdf?sfvrsn=20e389c6_12.
- ———. 2017b. "Individual Application for INCOSE ESEP." https://www.incose.org/docs/default-source/certification/esep-individual-application-form-41.pdf?sfvrsn=162b8bc6_18.
- Koch, A. 2020. "Shu Ha Ri: The Japanese Way to Improve Excellence." *Makigami*, 4 April. https://www.makigami.info/continuous-improvement-tools/shuhari-japanese-learning-system/.
- Lussier, J. W., and S. Shadrick. 2003. "Adaptive Thinking Training for Tactical Leaders." Paper presented at the RTO HFM Symposium, Genoa, Italy, 13-15 October.
- Polanyi, M. 1966. *The Tacit Dimension*. Chicago, US-IL: University of Chicago.
- Robinson, F. 2019a. "Exploring the Core of Systems Engineering." *SSRN Electronic Journal*. https://dx.doi.org/10.2139/ssrn.3339087.
- ———. 2019b. "Exploring the Core in Systems Engineering." Paper presented at the 22nd Annual Systems & Mission Engineering Conference, Tampa, US-FL, 21-24 October.
- Rother, M. 2016. *Phases of Practice: Shu Ha Ri*. YouTube. https://www.youtube.com/watch?v=0d6fndfodxU.
- Takamura, Y. 2004. "Teaching and Shu-Ha-Ri." *Dojo Blog*, 31 October. https://www.kimusubiaikido.com/blog/teaching-and-shu-ha-ri.
- Trogstad, I. K., S. Kokkula, and J. Aker. 2021. "Application of T-shaped Engineering Skills in Complex Multidisciplinary Projects." *INCOSE International Symposium* 31 (1): 1113-1129. https://doi.org/10.1002/j.2334-5837.2021.00890.x
- Yuta, K. 2021. "Common Misunderstandings of Shu Ha Ri Model." *Kosoado Japan*, 8 January. https://kosoadojapan.com/shu-ha-ri-model.

ABOUT THE AUTHOR

Fred Y. Robinson is a principal systems engineer who has worked for the MITRE Corporation for the last decade, having prior work experiences for similar lengths at Lockheed Martin and Fairchild Space/Controls. Fred began his engineering career with a Bachelor of mechanical engineering, later achieving an MS in systems engineering, and recently receiving a Doctor of Science in information systems & communications where he researched the impact of a unique communication challenge, the Jingle Fallacy, in systems acquisition. He has worked as both systems engineer & system architect leveraging systems thinking within/across multiple business and technology domains, including but not limited to census data capture, electronic digital archives, physical security, biometrics, healthcare, defense, and treasury systems/programs. Lockheed Martin recognizes Fred as a Certified Systems Architect. His past experiences also include product support, test engineering, and execution for spacecraft & aviation systems/components.

What Is the Role of a Systems Engineer In an Engineering Organization?

Richard Beasley, richard.beasley@rolls-royce.com

ABSTRACT
This article discusses the role of a "specialist" Systems Engineer inside an engineering focused on ensures the engineering parts integrate to achieve the objectives of the whole – and so is embedding a systems approach throughout the organization – trying to "make Systems Engineering the way engineering is done". In this type of organization all engineers need systems engineering as a core skill (which is part of them becoming T-shaped". The specialist Systems Engineer needs to be more π-shaped, with specialism in the systems approach used to inform and guide all the other disciplines which need to be integrated together. Since systems engineering is an integrating discipline the group of systems engineers must not become "just another" technical silo.

Systems Engineering aims to ensure that the pieces of a system work together to achieve the objectives of the whole (INCOSE 2022). Studies show that Effective Systems Engineering increases the probability of successful development of a system (Elm and Goldensten 2012). This is true for both, engineered systems and the system that is used to engineer the systems the organization produces. To effectively engineer systems the pieces of the organization need to fit together into an effective "realisation system," see Figure 1 (Beasley and Pickard 2020). The enterprise level realisation system is a complex adaptive system, with at least as many (if not more) difficulties, challenges and complexity as the systems that are produced.

A common view of Systems Engineering roles is that they act as the glue or integration between other engineering roles (Sheard 1996). It is the author's contention that Systems Engineering is a discipline that an organization cannot add as another discipline or silo into a pre-existing engineering organization. Systems Engineering is an integrating discipline, pulling together the more "traditionally" specialist engineering roles. By itself, Systems Engineering does not produce successful systems – there must be the other engineering disciplines as well. Therefore, in the author's company, the long-term objective is "*to make Systems Engineering the way we do engineering*." This means that the "existing" engineers must be able to understand and participate in Systems Engineering activities. As a result, they provide introductory Systems Engineering and Systems Thinking training to all new graduate engineers entering the company and offer it to all existing engineers. Over 2,000 engineers have had Systems Engineering training in the company over the past 15 years.

This vision faces two important challenges/questions:

1. How do we improve the state of Systems Engineering practice across the whole organization?

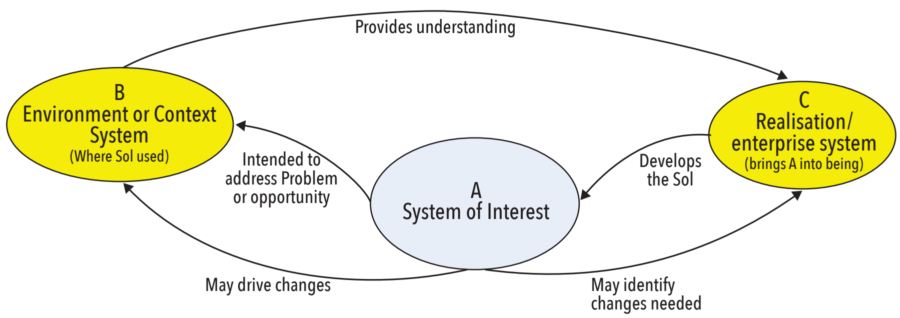


Figure 1. Three systems involved in developing and delivering product (Beasley et al, 2020)

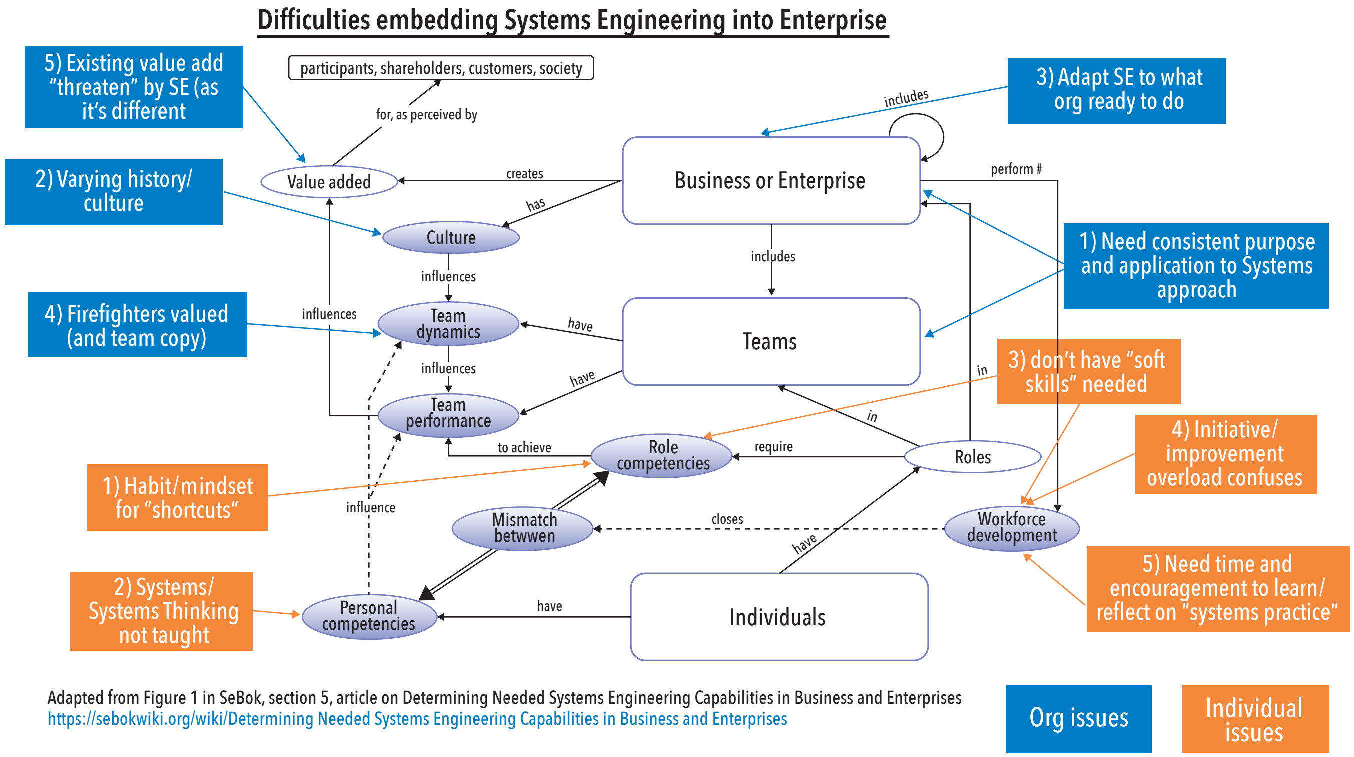


Figure 2. Summary of issues faced implementing systems engineering (adapted from Figure 1 in SEBoK, section 5, article on Determining Needed Systems Engineering Capabilities in Businesses and Enterprises (SEBoK Editorial Board, 2021), from Dunford et al, 2021

2. If everyone is doing Systems Engineering, what is the role for a "specialist" Systems Engineer?

Section 5 of the SEBoK (SEBoK 2021) discusses what is needed in an organization to do Systems Engineering, and Figure 2 (Dunford, Beasley, and Palmer 2021) illustrates the range of problems mapped onto what is considered to be needed in an organization that does Systems Engineering. These problems divide into organization and individual issues. Several of the problems come from the nature of traditional engineering activity – which places strong emphasis and great reward on fighting fires rather than problem prevention. (Beasley, Pickard, and Nolan 2014).

Researchers have explored the problem of implementing Systems Engineering into Rolls-Royce (Dunford et al. 2013), which shows advances are needed in leadership, application, and expertise of Systems Engineering. Figure 3 represents the organization as a dynamic system, and improvement of Systems Engineering comes from leadership pull, meaningful application, and reflection on the application to develop and expand expertise.

This implies that, as stated in the vision "make Systems Engineering the way we do Engineering" all engineers in addition to their prime skill, need Systems Engineering. Rolls-Royce does through the organization processes embedding, Systems Engineering processes, and training all Engineers in Systems Engineering / Systems Thinking. This helps make them "T-shaped" people. T-shaped is a metaphor to describe the "shape" of someone's skills – in this context the vertical line represents a depth of technical expertise, and the horizontal line the ability to collaborate across disciplines with experts in other areas. Some researchers argue that the "horizontal" collaboration includes the core systems engineering needed in "all engineering," providing skills (in addition to a deep engineering specialism) helping an approach to work together to ensure an optimised whole (rather than a part or single attribute). This leaves the question of the role of a more "specialist" systems engineer, which this article discusses below in terms of turning the T into a π-shaped engineer.

In Rolls-Royce the engineering teams generally look after either a specific system of interest (with the whole engine), one of the physical sub-systems (such as the combustor), the integrating "product

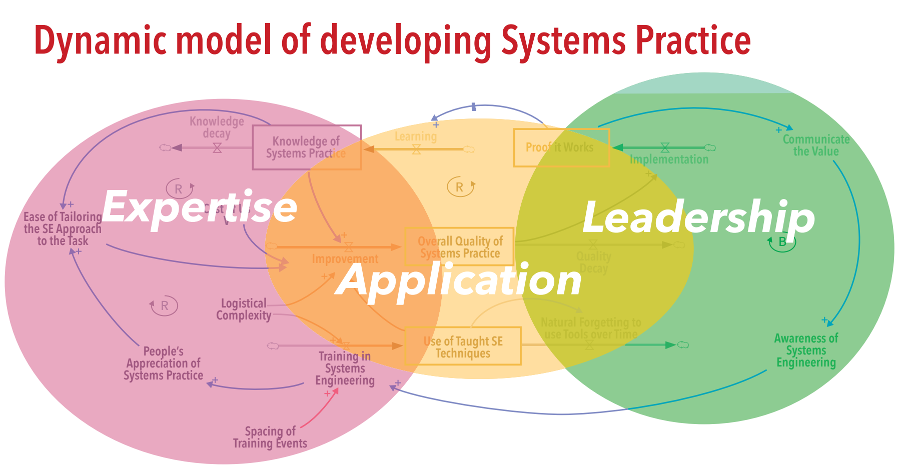


Figure 3. Spiral dynamics model (from Dunford et al, 2013)

system" (such as the engine secondary air cooling/sealing system, and oil system), specific components (a turbine blade), or an attribute (such as engine performance or weight) being the typical systems of interest). As well as the specific disciplines and technologies associated with their system of interest, it is important that they understand the insights that Systems Engineering gives in terms of context and understanding of the whole. This means it is considered important that the designers of a system of interest, those responsible for delivering it, are heavily involved in translating the system needs into requirements – they need to understand them and then deliver a solution that meets them. The specialist Systems Engineer provides a strong degree of specialist leadership in requirements elicitation and analysis. This emphasis on requirements should not result in the Systems Engineer being considered purely a "Requirements Engineer." Requirements are a part of, but not the complete scope, of Systems Engineering, and allowing the Systems Engineering to be seen as the sole "owners" and specialists in requirements would go against the "make Systems Engineering the way Rolls-Royce does Engineering" vision.

So: where does that leave the Systems Engineers?

There are two Systems Engineering roles in Rolls-Royce

1. Systems Engineering experts who have a deep specialism in the Systems Approach, Process, and Methods. People in this role provide expertise to support engineers in the project (via the Project Systems Engineers (described below). Additionally, they spend time developing Engineering/Systems Engineering capability (contributing to System C shown in Figure 1), and coaching/mentoring engineers to help develop the levels of expertise (the left-hand loop shown in Figure 3)
2. Project Systems Engineers (PSEs) embedded in project teams lead/facilitate the application of specific Systems Engineering practices in the project teams – both business and technical. This role involved being a full member of the project technical team and ensuring that effective Systems Engineering is both planned (working with the Program management) and executed. These aspects include:
   - facilitating Systems Thinking and Requirements analysis to understand the problem.
   - capturing/extending that understanding in Model-Based Systems Engineering
     - importantly, this modelling focuses on the enterprise level (using the Unified Architecture Framework (UAF), and working not only to understand end-user operation and the context for the Rolls-Royce product but to influence the end user solution with the Rolls-Royce power system capability
   - Taking the modelling into the definition of valid and complete requirements, solution architecture, and verification plans.
   - management/organization of the product data.
   - Ensuring that activities address the impacts of maturation of understanding.

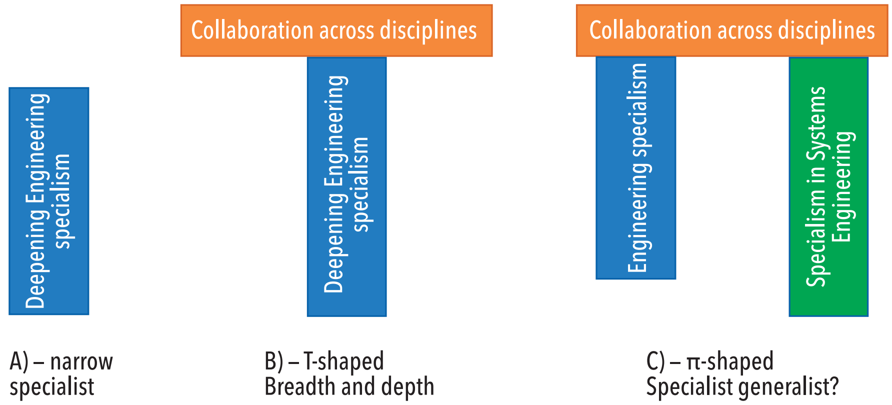


Figure 4. Narrow, T, and π-shaped engineers

These roles fulfil the Facilitator and Expert activities described in Annex C of the Systems Engineering Competency Framework (INCOSE 2018).

The Project Systems Engineer is the important interface between the product, technical engineering, and the program management/business layer. The important overlap between Systems Engineering is well recognized, (INCOSE UK 2020) is one of many examples, and the overlapping activities described in output from joint international working group with the Project Management Institute (PMI) (Rebentisch 2017). This does not make the Project Systems Engineer a variant of a Program Manager or diminish their technical engineering status – although it does open a pathway to a future career in Program Management or as a Project Executive. At its simplest, this responsibility is to make sure that the planning and organization of the System Development work recognises and includes the appropriate activities needed by the Systems Approach, and activities required by the uncertainties and risks identified by those activities. However, there is much more than that. A key aspect of a Systems Engineering approach is the recognition and integration of the needs of all the stakeholders for any given system of interest. For a technical product (such as what Rolls-Royce makes) it is too easy to focus only on the technical needs of the end users and the platform into which the Rolls-Royce power system goes. The Rolls-Royce business is a key stakeholder in the requirements for the product.

The Product Systems Engineer role is involved in helping to understand the business layer, and extracting the business stakeholder needs from that layer, to provide coordinated and consistent flow down into the product, service, and programme requirements. Once completed, there is still the work of recognising emergent technical risk as the product development matures understanding of the problems, including identification of assumptions and emerging concern.

In summary, this makes a Systems Engineer special, within the Engineering team. The Systems Engineering roles require "π-shaped" people – illustrated in Figure 4.

Being π-shaped implies that those in explicitly Systems Engineering roles have a technical engineering specialism and the connection between disciplines implicit in a "T-shaped" engineer, but they add a deeper Systems Engineering expertise and mindset (to enable the engineering teams to do Systems Engineering). Systems engineers become specialist generalists capable of working with a wide range of specialists and across a range of domains. The specialism becomes a focus on "how" engineering is done, and focused on integrating the technical silos, and ensuring/helping them to work together.

It is important to note that the specialist systems engineers need to remain technical engineers. That technical engineering element of a systems engineer is vital for three reasons

a) Credibility with the other engineers

b) Ability to interpret/understand the technical details coming from the specialist engineers
c) Ability to understand the insights/ issues identified in Systems Engineering and communicate these (as "the next question") back to the rest of the engineers, and the program management areas so that all gain and use the insight coming from the Systems Engineering pre-work that prevents problems.

Systems Engineering provides a vital input to ensure that product development is successful. Good Systems Engineering practice increases the probability of successful system development – allied to all the "traditional" disciplines of technical engineering. The specific challenge for Systems Engineers is that Systems Engineering is a transdisciplinary and integrative approach, and so different from the traditional, more "siloed" engineering disciplines. ■

REFERENCES

- Beasley, R., A. Pickard, and A. Nolan. 2014. "When "Yes" is the Wrong Answer." Paper presented at the 24th Annual International Symposium of INCOSE, Las Vegas, US-NV, 30 June–3 July.
- Beasley, R., and A. Pickard. 2020. "The Capability to Engineer Systems is a System Itself!" *INCOSE International Symposium* 30 (1): 1153-1168. https://doi.org/10.1002/j.2334-5837.2020.00778.x
- Dunford, C., R. Beasley, and E. Palmer. 2021. "Social Science Solutions for the Systems Engineer: What's Needed?" *INCOSE International Symposiu*m 31 (1): 699-712. https://doi.org/10.1002/j.2334-5837.2021.00863.x
- Dunford, C. N., M. Yearworth, D. M. York, and P. Godfrey. 2013. "A View of Systems Practice: Enabling Quality in Design." *Systems Engineering* 16 (2): 134-151. https://doi.org/10.1002/sys.21220
- Elm, J., and D. Goldenson. 2012. "The Business Case for Systems Engineering Study: Results of the Systems Engineering Effectiveness Survey" http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=34061.
- INCOSE. 2018. "Systems Engineering Competency Framework." https://www.incose.org/products-and-publications/competency-framework.
- INCOSE. 2021. "Systems Engineering Vision 2035." https://www.incose.org/about-systems-engineering/se-vision-2035.
- INCOSE UK. 2020. "Z11 Systems Engineering and Program Management." Z11_issue1.2_2020update_affinity_file (incoseuk.org).
- Rebentish, E. 2017. *Integrating Program Management and Systems Engineering: Methods, Tools, and Organizational Systems for Improving Performance*. Hoboken, US-NJ: Wiley.
- SEBoK. 2021. "Enabling Systems Engineering." https://www.sebokwiki.org/wiki/Enabling_Systems_Engineering#:~:text=Part%205%20of%20the%20Guide,described%20elsewhere%20in%20the%20SEBoK.
- Sheard, S. A. 1996. "Twelve Systems Engineering Roles." Paper presented at the Sixth Annual International Symposium of INCOSE, Boston, US-MA, 7-11 July.

ABOUT THE AUTHOR

Richard Beasley joined Rolls-Royce in 1986 with a Physics Degree from Bristol University, and an MSc in Gas Turbine Engineering from Cranfield University. After working on Integration Aerodynamics, Safety, Reliability and Life Cycle Engineering, he became Global Chief of Systems Engineering. In 2011 he became Rolls-Royce Associate Fellow in Systems Engineering. He was part of the BKCASE SEBoK author team, a leading author on the INCOSE SE competency framework, is a Past-President of the UK INCOSE Chapter, and is currently Services Director for INCOSE. He is a Chartered Engineer, Fellow of the Royal Aeronautical Society, INCOSE ESEP, and was a Visiting Fellow to the Systems Centre at Bristol University.

SEPTEMBER 2022
VOLUME 25/ ISSUE 3

Systems Skills…From Here to Diversity

Alan Harding, alan.harding@incose.net

■ **ABSTRACT**
Competency, the ability to do things, is at the heart of how systems engineers realise successful systems. Over the years INCOSE and partners have codified what this means, initially in the INCOSE UK Competency Framework (INCOSE UK 2010) later adopted globally by INCOSE and used as the basis of the INCOSE Systems Engineering Competency Framework (INCOSE 2018) which, notably, included a new area of professional skills. This article considers competency from the perspective of Diversity, Equity, and Inclusion (referred to as DEI) using a variety of sources including the recently published INCOSE SE Vision 2035 (INCOSE 2022) and a variety of competency frameworks to offer a view of how the skills and competencies of systems engineers need to evolve in the future. Five new competencies are proposed, as are opportunities to improve the definitions of five more.

1. INTRODUCTION

It is now accepted that a well-led diverse team, where every member is included, respected, and valued, delivers improved results compared to an equally well-led team that is either less diverse or less inclusive. These results may be either in business terms like profit or in terms of an outcome or system that is more inclusive and hence better suited to its full target audience (Harding and Pickard 2019).

Systems engineers enable the development and sustainment of successful systems, working effectively across the disciplines and engaging widely with stakeholders. Arguably systems engineers are already diverse in outlook, but of course, improvement is always possible. Historically this diverse outlook has spanned the technical disciplines and application domains of systems engineering. In recent years the need to understand the social dimension of systems engineering has been recognised, especially since many systems, and arguably all systems of systems are socio-technical in nature.

Additionally, the context where systems engineering is practiced continues to evolve. Harding and Pickard further recommended that "systems engineering workforce and culture should be at the forefront of diversity and inclusion."

This article explores what is already said about how the abilities of systems engineers should contribute to diversity and inclusion, and it also reflects on what more needs to be said based on the current in social media and other forums. The article will discuss:

- Definitions
- Diversity as highlighted in the INCOSE Systems Engineering Vision 2035
- Diversity in the UNESCO Engineering Report 2021
- Insights from current competency frameworks
- Insight from wider sources on Diversity
- Summary
- Conclusions

2. DEFINITIONS

INCOSE has adopted a set of definitions of Diversity, Equity, and Inclusion based on those established by ABET (ABET 2021), which are:

- Diversity is the range of human differences, encompassing the characteristics that make one individual or group different from another.
- Equity is the fair treatment, access, opportunity, and advancement for all people, achieved by intentionally focusing on their disparate needs, conditions, and abilities.
- Inclusion is the intentional, proactive, and continuing efforts and practices in which all members respect, support, and value others.

The successful application of these concepts throughout INCOSE and in the wider systems engineering perspective is intended to ensure that every person is comfortable being themselves and fully engages in leadership and all activities consistent with their individual talents, preferences, limitations, and ambitions.

Bringing this notion of "every person" to life, Figure 1 gives a view of the range of characteristics that can differ in each person. The author contends that even the apparently "pale, male and stale" cliché of a privileged and un-diverse white male (coined by NASA administrator Daniel Goldin in 1992) can mask individual characteristics such as caring responsibilities, neurodiversity, invisible health conditions or disabilities, or introversion which makes each of us unique. Hence, this discussion of skills for inclusion can be applied to any team or collective activity and for all situations where a system is being realised.

Reflection on the dimensions of diversity shown in Figure 1 highlights the need for

Figure 1. Categorised dimensions of diversity (based on SEBoK original [Ref. 3], adapted from (Harding and Pickard 2019))

competencies to address diversity and inclusion as shown in Table 1. In this and the following tables, the mapping is to existing competencies in the INCOSE Systems Engineering Competency Framework (ISECF), together with:

- Proposed New competencies
- Opportunities to improve an existing competency

3. DIVERSITY IN THE SYSTEMS ENGINEERING VISION 2035

The INCOSE Systems Engineering Vision 2035 (INCOSE 2022) is the product of sustained collaborative activity with inputs from across the industry, academia, and government. One of the uses of the vision is to support collaborative efforts to advance the discipline and grow the skill base to meet current and future challenges related to systems development.

This article has reviewed how diversity, equity/equality, and inclusion are represented in the Vision (plus the closely related social aspects) and what insights into systems engineering competencies we can take from this. Figure 2 and Figure 3 shows what was identified by this analysis.

Consideration of Figure 2 and Figure 3 highlights the need for the competencies shown in Table 2 (next page), including four potential new competencies or areas of knowledge.

4. DIVERSITY IN THE UNESCO ENGINEERING REPORT 2021

The recent report "Engineering for Sustainable Development (UNESCO 2021) includes section 2.1 covering Diversity and Inclusion in Engineering.

This notes that "the inherent skills required from engineers are distinctly changing … the need for people with competencies that were previously described as 'soft skills' are increasingly being seen as the 'critical skills' of the future." The INCOSE Systems Engineering Competency Framework, as discussed earlier, recognises this with the inclusion

Table 1. Consideration of dimensions of diversity

Observation from the Dimensions of Diversity	Related ISECF Competency
All systems engineers must have strong communication skills to enable them to engage empathetically with the widest possible range of stakeholders and colleagues.	Communications Emotional Intelligence
We must ensure that all systems engineers have a good understanding of the dimensions of diversity, how they affect peoples' preferences, and to inform how we interact with each other.	**New:** DEI Awareness
We must ensure that all systems engineers have a good understanding of sustainability with relevance to the social aspects which include environmental justice, human health and wellbeing, resource security, education, peace and political stability.	**New:** Sustainable Development Awareness (ecological, economic, social)
We must ensure that systems are conceived and realised that are inclusive for the widest possible stakeholder community.	**New:** Design for Inclusion (Inclusive Engineering)

Word search	Instances
• Diversity/diverse 16 mentions • Inclusive/inclusion 1 mention • Equity/equality 1 mention	• Diverse people/workforce/team (4) • Diverse stakeholders (4) • Diverse fields/domains (2) • Diverse spectrum of societal needs • Diverse set of solution alternatives • Diverse range of systems parameters (system health) • Demand for social equality

Figure 2. Mentions of diversity, equity, and inclusion in the Systems Engineering Vision

Word search	Instances
• Social equality – 23 mentions	• System/product social acceptability (5) • Changing global social environments (5) • STEM & Social sciences learning, curricula, skills (4) • Social needs (UN SDGs) (2) • Social responsibility and sustainability in system models • Non-engineered social and environmental systems • System social and ethical implications • Social equality • Social aspirations • Resolution of social problems • Increased reliance on social communication communities • PESTEL factors (incl. Social)

Figure 3. Mentions of social aspects in the Systems Engineering Vision

Table 2. Consideration of Systems Engineering Vision 2035

Competency	DEI	Social
Systems thinking (including PESTEL analysis)		Y
Requirement definition (understanding stakeholder needs)	Y	Y
Systems architecting (consideration of all viewpoints)	Y	Y
New: Design for Inclusion (Inclusive Engineering)	Y	Y
New: DEI Awareness	Y	Y
New: Sustainable Development Awareness (ecological, economic, social)	Y	Y
New: Social Science/Social Systems Engineering Awareness		Y

Table 3. Considerations from UNESCO report

Mentioned in UNESCO Report	Related ISECF Competency
Resilience (ability to adapt well in face of adversity or stress)	**Improvement**: Ethics and Professionalism
Agility (ability to react to change)	Planning
Ability to acquire new knowledge	Ethics and Professionalism
Team working	Team Dynamics
Communication	Communications

of a set of professional competencies.

The subject report continues to say that "Competencies such as resilience, agility, the ability to acquire new knowledge, team working and communication will all become as important, if not more important, than the detailed technical knowledge that has previously been valued in engineering." Table 3 summarises how these aspects relate to ISECF competencies.

5. INSIGHTS FROM CURRENT COMPETENCY FRAMEWORKS

The competency frameworks used to underpin this discussion are as follows:

1. INCOSE Systems Engineering Competency Framework
2. INCOSE Technical Leadership Model
3. UKSPEC - The UK Standard for Professional Engineering Competence and Commitment
4. Atlas: The Theory of Effective Systems Engineers

5.1 INCOSE SYSTEMS ENGINEERING COMPETENCY FRAMEWORK

The INCOSE Systems Engineering Competency Framework (ISECF) (INCOSE 2018) is the most widely recognized description of the knowledge, skills, and experience needed to perform systems engineering. This framework, based on prior work in the UK, is summarized in Figure 4. It comprises competencies described in five themes: core, professional, management, technical, and integrating.

This article uses this framework as the basis of its findings/discussion and will map other sources to it.

CORE COMPETENCIES

Core competencies underpin engineering as well as systems engineering.

Systems Thinking	The application of the fundamental concepts of systems thinking to systems engineering;
Lifecycles	Selection of the appropiate lifecycles in the realization of a system;
Capability Engineering	An appreciation of the role the system of interest plays in the system of which it is part of;
General Engineering	Foundational concepts in mathematics, science and engineering and their application;
Critical Thinking	The objective analysis and evaluation of a topic in order to form a judgement;
Systems Modeling and Analysis	Provision of rigorous data and informaion including the use of modeling to support technical understanding and decision making.

PROFESSIONAL COMPETENCIES

Behavioral competencies well-established within the Human Resources (HR) domain. To facilitate alignment with existing HR frameworks, where practicable, competency definitions have been taken from well-established, internationally-recognized definitions rather than partial or complete re-invention by INCOSE.

Communications	The dynamic process of transmitting or exchanging information;
Ethics and Professionalism	The personal, organizational, and corporate standards of behavior expected of systems engineers;
Technical Leadership	The application of technical knowledge and experience in systems engineering together with appropriate professional competencies;
Negotiation	Dialogue between two or more parties intended to reach a beneficial outcome where difference exist between them;
Team Dynamics	The unconscious, psychological forces that influence the direction of a team's behavior and performance;
Facilitation	The act of helping others to deal with a process, solve a problem, or reach a goal without getting directly involved;
Emotional Intelligence	The ability to monitor one's own and others' feelings and use this information to guide thinking and action;
Coaching and Mentoring	Development approaches based on the use of one-to-one conversations to enhance an individual's skills, knowledge or work performance.

MANAGEMENT COMPETENCIES

The ability to perform tasks associated with controlling and managing Systems Engineering activities. This includes tasks associated with the Management Processes identified in the INCOSE SE Handbook.

Planning	Producing, coordinating and maintaining effective and workable plans across multiple disciplines;
Monitoring and Control	Assessment of an ongoing project to see if the current plans are aligned and feasible;
Decision Management	The structured, analytical framework for objectively identifying, characterizing and evaluating a set of alternatives;
Concurrent Engineering	A work methodology based on the parallelization of tasks;
Business and Enterprise Integration	The consideration of needs and requirements of other internal stakeholders as part of the system development;
Acquisition and Supply	Obtaining or providing a product or service in accordance with requirements;
Information Management	Addresses activities associated with all aspects of information, to provide designated stakeholders with appropriate levels of timeliness, accuracy and security;
Configuration Management	Ensuring the overall coherence of system functional, performance and physical characteristics throughout its lifecycle;
Risk and Opportunity Management	The identification and reduction in the probability of uncertain events, or maximizing the potential of opportunities provided by them.

TECHNICAL COMPETENCIES

The ability to perform tasks associated primarily with the suite of Technical Processes identified in the INCOSE SE Handbook.

Requirements Definition	To analyze the stakeholder needs and expections to establish the requirements for a system;
System Architecting	The definition of the system structure, interfaces and associated derived requirements to produce a solution that can be implemented;
Design for…	Ensuring that the requirements of all lifecycle stages are addressed at the correct point in the system design;
Integration	The logical process for assembling a set of system elements and aggregates into the realized system, product, or service;
Interfaces	The identification, definition and control of interactions across system or system element boundaries;
Verification	The formal process of obtaining objective evidence that a system fulfills its specified requirements and characteristics;
Validation	The formal process of obtaining objective evidence that a system achieves its intended use in its intended operational environment;
Transition	Integration of a verified system into its operational environment including the wider system of which it forms a part;
Operation and Support	When the system is used to deliver its capabilites, and is sustained over its lifetime.

INTEGRATING COMPETENCIES

This competency group recognizes Systems Engineering as an integrating discipline, joining activities and thinking from specialists in other disciplines to create a coherent whole.

Project Management	Identification, planning, and coordinating activities to deliver a satisfactory system, product, service of appropriate quality;
Finance	Estimating and tracking costs associated with the project;
Logistics	The support and sustainment of a product once it is transitioned to the end user;
Quality	Achieving customer satisfaction through the control of key product characteristics.

Figure 4. Complete listing of competencies in the INCOSE Systems Engineering Competency Framework

Table 4. Consideration of INCOSE Technical Leadership Model

Leadership attribute	Related ISECF Competency
Thinks strategically – includes considering whole range of stakeholders who benefit and are impacted by the full lifecycle of the product. Opportunity to improve definition by highlighting the diverse range of stakeholders who both benefit and are impacted by the system development.	**Improvement:** Technical Leadership
Fosters collaboration – collaboration across and within diverse groups.	Technical Leadership
Communicates effectively – ability to communicate and influence in language suited to diverse groups.	Communications Facilitation
Enables others to be successful – the ability to recognize potential and to enable all members of the team to be successful, including those traditionally overlooked or who do not assert themselves for cultural or personal reasons.	Coaching and mentoring
Demonstrates emotional intelligence – importance is increased when engaging with diverse groups of stakeholders, encountering different cultures, expectations, working practices, and more. Opportunity to improve definition by broadening definition to cover self-awareness, self-regulation, motivation, empathy, and social skills.	**Improvement:** Emotional Intelligence (Goleman 1996)

In the main, two questions will be addressed:

1. Which are the key competencies that enable systems engineers to foster and operate well in diverse situations (and to realise inclusive products)?
2. Are there other competencies or refinements to what we have that would help systems engineers achieve even more?

Within the detailed descriptions of these competencies, only one explicitly calls out diversity as intended in this article. This is the "Emotional Intelligence" competency whose description notes that "Systems Engineering involves interacting with many diverse stakeholders." Elsewhere a "diverse range of projects" is referred to, but this relates to a systems engineer having to understand and contribute to various projects.

5.2 INCOSE TECHNICAL LEADERSHIP MODEL

The INCOSE Technical Leadership Model (Godfrey 2016) provides a view of what is required to be a leader in systems engineering, in addition to the practitioner skills required. Six interrelated attributes are identified in the model shown in Figure 5.

The attributes highlighted in green have a direct bearing on diversity and inclusion and are described in Table 4.

5.3 THE UK STANDARD FOR PROFESSIONAL ENGINEERING COMPETENCE AND COMMITMENT

The UK Standard for Professional Engineering Competence and Commitment (Engineering Council 2020) commonly known as UKSPEC, defines the competence and commitment requirements for people to be Professionally Registered as Engineers in the UK, giving a wider view of engineering professional competencies.

While this is a UK framework, it is widely recognised internationally and is also typical of professional engineering frameworks from other nations.

Figure 6 shows the five UKSPEC competencies. The competency areas highlighted in green have a direct bearing on diversity and inclusion, as shown in Table 5 and expanded below.

- B1 – "Take an active role in the identification and definition of project requirements, problems and opportunities" – engaging with and understanding the viewpoints and needs of all stakeholders, including those diverse stakeholders who have not traditionally been considered.
- C1 – "Communicate effectively with others, at all levels, in English" – ability to communicate and influence in language suited to diverse groups. This may be in whichever language(s) are required and includes the need to communicate with people who have differing needs to enable comfortable and effective engagement.
- C3 – "Lead teams or technical specialisms and assist others to meet changing technical and managerial needs" – the ability to recognize potential and to enable all members of the team to be successful, including those traditionally overlooked or who do not assert themselves for cultural or personal reasons.
- D3 – "Demonstrate personal and social skills and awareness of diversity and inclusion issues" – includes emotional intelligence; creating, maintaining, and enhancing productive working relationships, resolving conflicts; and understanding and supporting the needs and concerns of others.
- E5 – "Understand the ethical issues that may arise in their role and carry out their responsibilities in an ethical manner" – includes applying awareness of the United Nations Sustainable

Figure 5. Attributes of a systems leader (INCOSE Technical Leadership Institute) redrawn for clarity

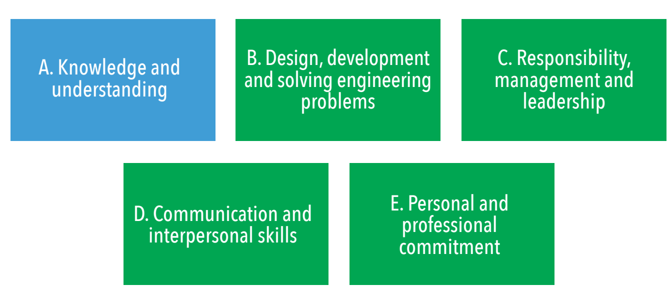


Figure 6. UKSPEC competencies A-E

Table 5. Consideration of UKSPEC competencies

UKSPEC competency area	Related ISECF Competency
B1 "Take an active role in the identification and definition of project requirements, problems and opportunities"	Systems Thinking Technical Leadership Requirements Definition
C1 "Communicate effectively with others, at all levels, in English"	Communications Emotional Intelligence
C3 "Lead teams or technical specialisms and assist others to meet changing technical and managerial needs"	Technical Leadership Team Dynamics Coaching and Mentoring
D3 "Demonstrate personal and social skills and awareness of diversity and inclusion issues"	Communications Emotional Intelligence **New:** DEI awareness
E5 "Understand the ethical issues that may arise in their role and carry out their responsibilities in an ethical manner". Opportunity to improve definition for instance includes applying awareness of UN Sustainable Development Goals, social justice, diversity equity and inclusion.	**Improvement:** Ethics and Professionalism

Table 6. Consideration of Atlas proficiency areas

Atlas Proficiency Area	Related ISECF Competency
1. Math/Science/General Engineering	No direct relevance to DEI
2. System's Domain & Operational Context	No direct relevance to DEI
3. Systems Engineering Discipline	No direct relevance to DEI
4. Systems Engineering Mindset • Big-picture thinking • Paradoxical mindset • Adaptability • Abstraction • Foresight and vision	Systems Thinking **New:** Adaptability
5. Interpersonal Skills • Communication • Listening and comprehension • Working in a team • Influence, persuasion, and negotiation • Building a social network Opportunity to improve definition of "Communications" by referring to diverse audiences, listening and comprehension, and building and maintaining professional/social networks. Opportunity to improve definition of "Negotiation" by also referring to influence and persuasion.	**Improvement:** Communications Team Dynamics **Improvement:** Negotiation
6. Technical Leadership • Building and orchestrating a diverse team • Balanced decision making and rational risk taking • Guiding diverse stakeholders • Conflict resolution and barrier breaking • Business and project management skills • Establishing technical strategies • Enabling broad portfolio-level outcomes Opportunity to improve definition of "Team Dynamics" by referring to diversity and inclusion within teams; and we should consider splitting out conflict resolution as it does not only exist within a team.	**Improvement:** Team Dynamics Systems Thinking Communications

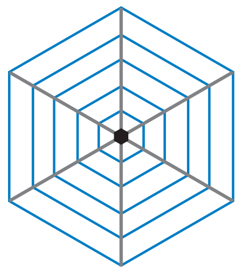


Figure 7 Atlas proficiency areas for systems engineers

Development Goals, social justice, diversity equity, and inclusion. This social dimension is growing in importance to systems engineers because often systems are socio-technical and may even include a political aspect.

As well as the specific competencies, the requirement for professional commitment within UKSPEC also calls for registrants to "show that they have adopted a set of values and conduct that maintains and enhances the reputation of the profession" in areas including recognising inclusivity and diversity.

5.4 ATLAS: THE THEORY OF EFFECTIVE SYSTEMS ENGINEERS

The US SE Research Center Helix program created the Atlas model (Hutchison et al. 2018) of "general principles and ideas that relate to the subject of what makes systems engineers effective and why." It talks in terms of the proficiencies of systems engineers, where proficiency is defined as "the quality or state of knowledge, skills, abilities, behaviors, and cognition." Figure 7 shows the six Atlas proficiency areas.

Reflection on the Atlas proficiency areas shown in Figure 7 highlights the need for competencies to address diversity and inclusion as shown in Table 6. The Atlas Proficiency Area number 4 is a key aspect for systems engineering in a diverse and changing context and is a strong contender to pull through from Atlas into ISECF, noting the Atlas definition: "The overall ability to deal with ambiguity and uncertainty, this involves the abilities to be open-minded, understand multiple disciplines, deal with challenges, and the ability to take rational risks."

6. INSIGHTS FROM WIDER SOURCES ON DIVERSITY SKILLS

Research for this article has included wider reading including on social media, and discussion with colleagues familiar with the field of DEI. Although not comprehensive, this has given a good sense of what is being highlighted as valuable when building and operating teams that are inclusive and hence are environments for a diverse range of individuals to thrive and give of their best.

- Systems Thinking – given the range of differences that people may have from one another (see Figure 1) both systems thinking and perspective taking are vital tools to understand and make sense of situations. It is also important to recognise and test any assumptions made, as these could risk stereotyping people.
- Empathy – much stress is given to the importance of empathy (the ability to understand and share the feelings of another) and both having and showing genuine interest in others to learn something new from them and build open and trusting relationships.
- Culture - it is clearly important to recognise cultural differences and to understand how to work together appreciating and recognising these differences in groups as well as in individuals. These could be aspects such as recognising different national and religious holidays, traditions around alcohol and mixing between sexes, or aspects such as differences in decision-making, leading, and communicating (Meyer 2014). This must be done, of course, while avoiding stereotyping people.
- Communications – it is vital to communicate inclusively, for instance listening with full attention, giving space and time to those who need longer to express themselves (those speaking in languages other than their own, those with speech/hearing difficulties, those needing a translator or person to sign for them). Also choosing a style, tone, and language to be as helpful as possible to the audience such as preferring simple language to jargon, speaking slowly as a courtesy to those less familiar with the language, and offering simultaneous translation/sub-titles to online audiences. It is also important to understand cultural differences regarding communication (women not wanting to challenge authority, different cultures not habitually asking questions).
- Psychological Safety - the need to ensure psychological safety within the team, thus allowing everyone to "be able to show and employ oneself without fear of negative consequences of self-image, status or career" (Kahn 1990). This includes confronting bias, any form of exclusion or intimidation, and mitigating microaggressions which can require significant courage.
- Delegation, Coaching and Mentoring – various sources highlight the positive contribution to inclusion that effective delegation, as well as coaching and mentoring can provide. Taking the chance to empower a less obvious candidate, perhaps combined with coaching and support could make a significant change to both an individual and the team dynamics. All feedback should be honest and fair and needs to draw heavily on the cultural awareness mentioned earlier to be received constructively. Mentoring can also be two-way, for instance in reverse mentoring methods.
- Meetings – it is stressed how important it is to run inclusive meetings, ensuring a fair chance for all to express themselves and managing the agenda to allow appropriate amounts of discussion. This is a way of accommodating individuals with markedly different seniority and status, communications styles, and confidence levels. This can be made easier or harder when using online communications which have become so prevalent during the Covid pandemic.

7. SUMMARY

This article has reviewed a range of sources on diversity and inclusion and has identified many insights into the competencies that systems engineers will need now and in the future. This section draws those insights together, mapping them back to the INCOSE Systems Engineering Competency Framework as shown in Figure 8, where competencies are labelled as follows:

- New (labelled as new) – additional competency suggested by the article.
- Improvement (labelled with an "I") – competency is relevant to diversity and inclusion and the article has identified an opportunity to improve its definition.
- Relevant (outlined in green) – competency is relevant to diversity and inclusion with no change.
- The remaining competencies have not been considered as particularly relevant to diversity and inclusion in this article.

The key conclusions of this article are as follows:

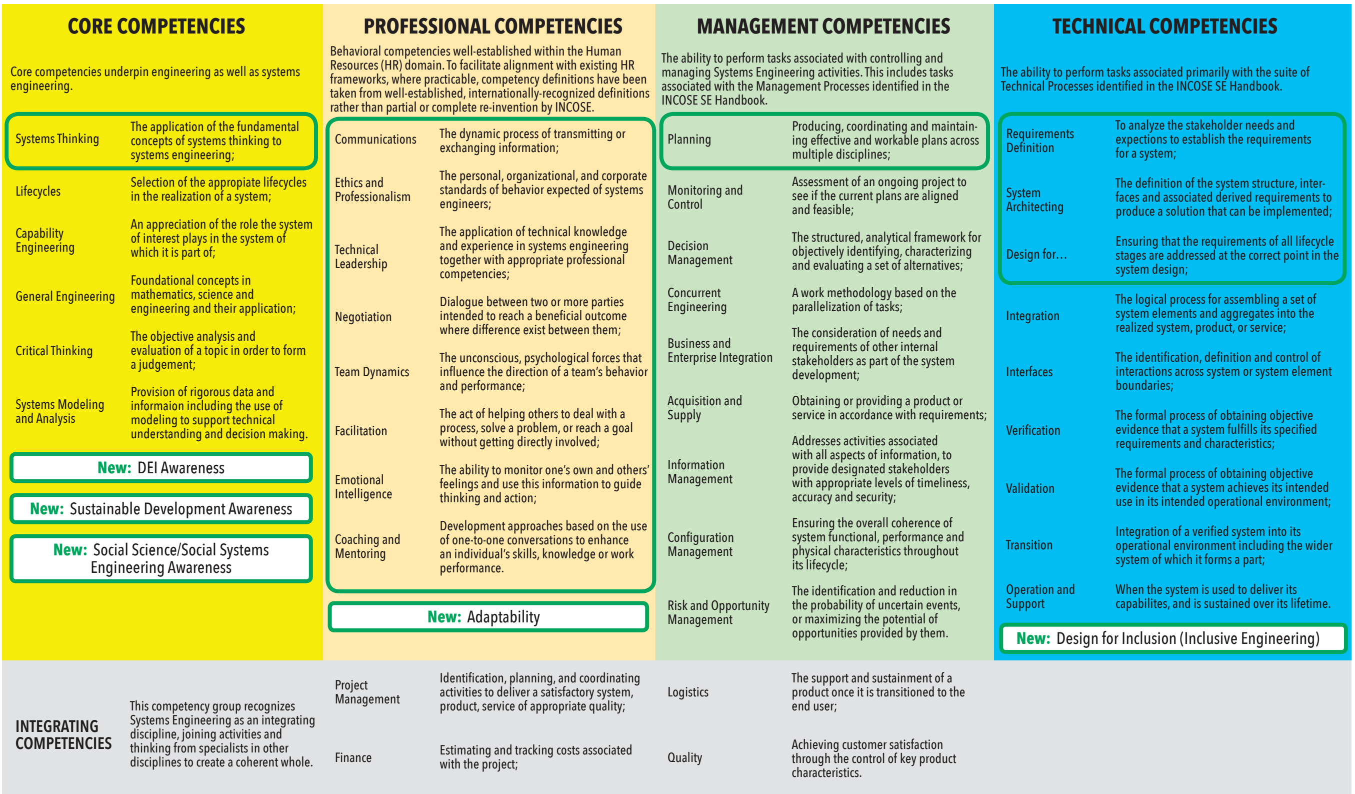


Figure 8 Modifications to the INCOSE Systems Engineering Competency Framework identified by this article

Table 7. New and improved competencies

New Competencies	Improved Competencies
DEI Awareness	Ethics and Professionalism
Sustainable Development Awareness	Technical Leadership
Social Science/Social Systems Engineering Awareness	Negotiation
Adaptability	Team Dynamics
Design for Inclusion (Inclusive Engineering)	Emotional Intelligence

Core Competencies

- Systems Thinking is highly relevant to DEI because it imparts the ability to discern and reason about multiple systems, including socio-technical, cultural, political, and more.
- Three new core competencies have been identified, positioned as core because they should underpin both engineering and systems engineering in the future.

Professional Competencies

- All these competencies are relevant to DEI – this is not a surprise as these all relate to both the individual systems engineer and engaging/leading/serving other stakeholders.
- Insights identified in this article suggest the opportunity to improve definitions of five of these competencies, to better highlight what is needed to foster and operate in diverse and inclusive situations.

Technical Competencies

- This article has highlighted the relevance of the three technical competencies that sit "earlier" in a traditional system lifecycle
- Design for Inclusion (Inclusive Design) has been proposed as an addition to the "Design For …" competency to widen the mindset of systems engineers when helping realise systems for necessarily diverse stakeholder communities.
- While this article did not address it, the author recognises that validation, operation, and support are also likely to be affected by diversity and inclusion considerations.

Management Competencies

- The UNESCO report recommends that engineers will need to be more agile in reacting to change, which is mapped to the Planning competency.

Integrating Competencies

- No specific mention of these competencies in this analysis.

8. CONCLUSIONS

This article set out to offer a view of how the skills and competencies of systems engineers need to evolve in the future. The INCOSE Systems Engineering Competency Framework remains a good basis to describe what knowledge, skills and behaviours are needed by a systems engineer.

Based on the consideration of DEI factors in this article, Table 7 lists the new competencies that have been identified, and the opportunities to improve the definitions of five more. The competency area where most change is proposed is the "Professional" competencies, which seems reasonable given the interpersonal and social emphasis of DEI.

As a "sense-check," the most frequently mentioned competencies (existing and new) in this article are lister below in rank order. They provide a good summary of the critical areas where systems engineers, and those who lead and develop them, will need to focus on the future in order to fully complete the journey "from here to diversity:"

1. Communications
2. Requirements Definition
3. Technical Leadership, Emotional Intelligence, DEI Awareness (New)

REFERENCES

- ABET. 2021. "Diversity, Equity & Inclusion." https://www.abet.org/about-abet/diversity-equity-and-inclusion/.
- Engineering Council. 2020. *UK Standard for Professional Engineering Competence and Commitment (UK-SPEC) 4th edition*. London, UK: Engineering Council. https://www.engc.org.uk/EngCDocuments/Internet/Website/UK-SPEC%20fourth%20edition.pdf.
- Godfrey, P. 2016. "Building a Technical Leadership Model." *INCOSE International Symposium* 26 (1): 757-772. http://dx.doi.org/10.1002/j.2334-5837.2016.00191.x
- Goleman, D. 1996. *Emotional Intelligence: Why it Can Matter More Than IQ*. London, UK: Bloomsbury.
- Harding, A. D., and A. Pickard. 2019. "Towards a more Diverse INCOSE." *INSIGHT* 22 (3): 14-20. https://doi.org/10.1002/inst.12254
- Harding, A. D., and A. Squires. 2022. "Diversity, Equity, and Inclusion." SEBoK, 20 May. https://www.sebokwiki.org/wiki/Diversity,_Equity,_and_Inclusion.
- Hutchison, N., D. Verma, P. Burke, M. Clifford, R. Giffin, S. Luna, and M. Partacz. 2018. *Atlas 1.1: An Update to the Theory of Effective Systems Engineers*. Hoboken, NJ: Systems Engineering Research Center, Stevens Institute of Technology.
- INCOSE. 2018. "Competency Framework." https://www.incose.org/products-and-publications/competency-framework.
- ———. 2021. "DEI-100: Diversity, Equity, and Inclusion." https://www.incose.org/docs/default-source/policiesbylaws/policy_dei-100-_-diversity-equity-and-inclusion.pdf?sfvrsn=3ba98c6_6.
- ———. 2022. "Systems Engineering Vision 2035." https://www.incose.org/about-systems-engineering/se-vision-2035.
- INCOSE UK. 2010. "Systems Engineering Competencies Framework." https://incoseuk.org/Documents/Groups/UKChapter/SE_Competencies_Framework_Issue_3.pdf.
- Kahn, W. A., 1990. "Psychological Conditions of Personal Engagement and Disengagement at Work." Academy of Management Journal 33 (4): 692–724.
- Meyer, E. 2014. *The Culture Map: Breaking Through the Invisible Boundaries of Global Business*. New York, US-NY: Public Affairs.
- UNESCO. 2021. "Engineering for Sustainable Development." https://en.unesco.org/reports/engineering#:~:text=By%20presenting%20case%20studies%20and,energy%2C%20responding%20to%20natural%20hazards%2C.

ABOUT THE AUTHOR

Alan Harding is a past-president of INCOSE and INCOSE UK and is a systems engineer with over 37 years' experience. He is a Global Engineering Fellow with BAE Systems Digital Intelligence, a Chartered Engineer, a Fellow of the IET, and a Freeman of the Worshipful Company of Engineers. During 2020 Alan led the activity to establish INCOSE's first ever policy on Diversity, Equity and Inclusion, DEI-100.

Systems Engineers – Value Added Product Owners

Aswin Sukumaran Nair, Aswin.Nair@Carrier.com

■ **ABSTRACT**

Agile Methodology requires specialized roles like Product Owners to act as the bridge between the business and the product delivery teams. This calls for specific skills such as a customer-centric, design-thinking mindset and a good vision of the overall system, its capabilities, and the ability to provide the right information at the right time.

The primary duty of a product owner is to maintain a well-refined, prioritized backlog of work items. Systems Engineers are enabled with the right methods and tools to perform solution architecture, design synthesis, system verification, and validation. Systems Engineers have visibility into the overall system, interfaces between sub-systems, and external interface requirements. Enabled with the knowledge of Technical Processes, Systems Engineers are able to describe System Elements, their behavior, and interactions in the best possible detail. This allows them to ensure that functional and non-functional backlog items are defined in unambiguous adequate detail.

By applying their knowledge and expertise as mentioned above, Systems Engineers can effectively perform Product Owner duties and enables Agile teams to work efficiently to deliver the correct system increment at the end of each pre-defined time box.

OVERVIEW

Many organizations have already adopted or are in the process of adopting Agile as a standard method for product/service development and delivery. This requires specialized roles like Product Managers and Product Owners to define and support the building of products that meet customer needs. Product Managers and Product owners act as the bridge between the business and stakeholders and the product delivery teams. This requires specific skills such as having a customer-centric, design-thinking mindset and a good vision of the overall system, its capabilities, and the ability to enable cross-functional teams with the right information at the right time. Systems Engineers possess the traits and skills necessary to perform these responsibilities (INCOSE 2015).

Agile teams work on "Backlogs" or holding areas that describe features that address user needs and deliver business benefits. It is always crucial to maintain a well-refined backlog. Systems Engineers can consider both the business and technical needs of customers and can translate stakeholder needs into requirements that can be easily understood by engineering and other functional teams since they are enabled with the right methods and tools to perform solution architecture, design synthesis, and system verification and validation. This allows them to effectively manage stakeholder needs and decompose them into various levels of "Backlog" with the right amount of detail.

Agile teams are generally focused on delivering a feature increment. This means that they need to be enabled with the underlying system, communication platforms, interfaces, and such for seamless feature delivery. Systems Engineers have visibility into the overall system and interfaces between various sub-systems within the system and external interface requirements. This allows them to ensure that enabler and non-functional backlog items are properly defined – which sometimes general product owners and business analysts find difficult to do.

When multiple self-organized agile teams are trying to deliver a product in increments, coordination among these teams becomes vital. Product Managers and Product owners need to ensure that each agile team is delivering the right "ingredient" that rolls up into the overall system or product. Enabled with the knowledge of Technical Processes, especially with the Implementation process, Systems Engineers can describe System Elements, their behavior, and interactions in the best possible detail. This allows Agile teams to work independently and deliver a system increment at the end of each pre-defined time box.

The points listed above are only a few among many reasons why Systems Engineers can effectively handle Product Owner duties in an Agile ecosystem and deliver added value by applying their knowledge and expertise. The article discusses this in detail in the next few sections.

BEING "AGILE" DURING SYSTEM DEVELOPMENT

In the changing times, it is imperative that organizations need to be able to deliver solutions that are more efficient, cost-effective, and available at a faster pace than their competitors. Many organizations over time have adopted or are in the process of adopting lean methods built on Agile practices. These practices enable organizations to work with greater agility, deliver in a continuous fashion, and can incorporate changes better.

Changes are typically introduced mostly by the following causes, which arise in the due course of system development.

SHIFT IN MARKET TRENDS:

The traditional waterfall method of solution/system/product development is linear in nature and requires a clear establishment of expectations and deliverables from a project at the outset itself. This poses a challenge – the market evolves very rapidly, and the initial set of needs and requirements can quickly change over time – in many cases during the execution of the project itself. Since the waterfall method relies heavily on the completion of an earlier phase in its entirety, for the next phase to begin, the cost of introducing changes at later phases is quite high.

EVOLVING STAKEHOLDER NEEDS:

Waterfall methods also require initial agreements with stakeholders that are frozen during early stages of the project. During later phases of product development, when the initial samples evolve, and teams present to stakeholders during scheduled phase gate reviews, it is possible that there could be differences in opinion from stakeholders regarding how they envisioned the product to be in the first place.

While good systems engineering practices in the early stages of needs analysis and effective requirement elicitation processes can address this to a certain extent, technical limitations, performance considerations, and more can have a detrimental effect on the overall experience that the stakeholders have once the initial samples of the product are ready. At this stage, since most of the design considerations are identified and locked down, the amount of flexibility in making changes tends to be on the lower side.

INTRODUCTION OF DEFECTS IN LATER STAGES:

During the phased product development approach — in many cases, verification and validation of the product and systems also happen in a phased manner. This means that the entire solution or product is picked up for verification after the design and implementation of the design are final. This can lead to late identification of unintended behaviors, issues occurring during boundary conditions, integration issues, and such. Since the design at that point is finalized, the cost of fixing these issues is significantly higher.

A good application of technical Processes such as system analysis process, integration process, and verification process reduces this risk significantly. However, with distributed teams, teams operating with higher autonomy, dependency on simulations, inability to put together real-world test conditions, and such can still cause defects to be identified late in the process.

IMPROVEMENTS WITH AGILE

In the Agile world, there are different levels at which "backlog" items are constantly reviewed and refined — At the Business Portfolio Level, At the Product or System level, At project levels, and at Agile team levels. Product Managers, Product Owners, Systems Engineers, and Implementation teams come together during pre-defined periodic intervals and evaluate if there are any changes to the initial set of requirements identified. These changes could have originated based on shifts in market trends, and evolving needs based on stakeholder experience.

In addition to this, Agile teams conduct a demonstration of the system capability and state achieved to date — providing the stakeholders and other decision-makers valuable inputs regarding system behavior progression in real-time. This enables the team to assess, prioritize, triage, and assign team members to solve defects in the system, and to prevent potential issues that could arise in the future in a methodical manner.

These periodic reviews or "the refinement process" and demonstrations allow for all stakeholders to convene and assess the state of the system or product development. Based on the inputs coming in from various sources as mentioned above, changes required are added to the appropriate backlog levels. The changes funnel down to the subsequent levels where appropriate course correction occurs. This provides great flexibility in terms of incorporating change and ensuring continuous delivery of system increments with built-in quality checks.

ROLE OF PRODUCT MANAGERS AND PRODUCT OWNERS

It is evident that the key to ensuring agility in managing change is continuous review and refinement of requirements and backlog items at various levels. Product Managers and Product Owners play a vital role in enabling this process.

PRODUCT MANAGERS

Product Managers are responsible for defining the system or product landscape by detailing the features and requirements (Scaled Agile Framework 2021a). They consult and collaborate with multiple functions — including and not limited to end customers, sales, marketing, regulatory and compliance entities, engineering, manufacturing, and others. They analyze and synthesize customer needs, provide insights into the overall Solution Context, and establish a vision and roadmap for the product and the program. They are vital for enabling Agile Release Trains (a team of multiple Agile teams that are responsible for developing system increments) to deliver value continuously.

PRODUCT OWNERS

Product Owners are part of Agile Teams themselves. They are "customer proxies" for the agile teams, and they work with Product Managers, Systems Engineers, other stakeholders, and Product Owners from other agile teams (Scaled Agile Framework 2021b). They enable distilling features into smaller "stories" and help the agile team prioritize them. They are primarily responsible for maintaining the backlog for Agile Teams and ensuring that program priorities are considered while streamlining program execution.

APPLYING SYSTEMS ENGINEERING PRACTICES AND SKILLS TO PRODUCT OWNER DUTIES

Systems Engineers are enabled with knowledge of tools and processes that help in analyzing problems and opportunities that a system presents in its entirety. They can understand the big picture, and they observe and study how different elements in the system interact and change over time. They can assess an issue fully and can contribute towards developing solutions that are optimal by considering multiple aspects — including and not limited to - short- and long-term consequences of actions, impact on the system structure, impact of time delays, and emergent behaviors.

A Systems Engineer's unique skillset developed by consciously being a Systems Thinker allow them to be able to convert ambiguous problem statements into clear and precise work items for the team. They can establish an environment where different teams collaborate and are conducive to collective decision-making and conflict resolution. They can communicate effectively to Organizational leadership and facilitate decision-making by presenting solutions, alternatives, implications of decisions, and putting forward recommendations and actions. Let us look at how these skillsets apply to the duties and responsibilities of a Product Owner below.

PLANNING:

Product Owners have multiple duties in terms of Planning and Execution. They are part of the refinement process and ensure continuous refinement of the program backlog by providing information on technical implementation details, issues faced by the teams, risks, and changes in priorities. They ensure that there is adequate capacity allocated for completion of functional features, making quality improvements, closing technical debts, addressing non-functional requirements, and making appropriate prioritization and adjustments to the execution plan as needed.

Systems Engineers see the big picture and understand how different system elements interact with one another and with external systems. They have an extensive understanding of system interfaces, capabilities, life cycle, feedback loops, and cause-and-effect relationships that individual Agile teams may not be able to comprehend. They are well versed in the Business or Mission Analysis Process and Stakeholder Needs and Requirements Definition Process — allowing them to characterize the solutions space and define the capabilities of the system. This puts them in a unique position to facilitate and contribute to refinement sessions at each stage. As a result, they can add immense value as a Product Owner in refining Product Landscape, Features, and other program backlog items and prioritizing them in accordance with how they envision the system to evolve.

ELABORATION AND ACCEPTANCE OF WORK ITEMS:

Backlog items are drilled down into and elaborated on further for implementation into a work item called "User Story." *Product Owners* ensure that the elaboration of user stories is complete, unambiguous, well defined, and the criteria for the user story to be "accepted" are well defined. They also have constant reviews of the stories with the agile teams that implement the user stories in "backlog refinement sessions." Product Owners also "accept" user stories by validating that they meet the acceptance criteria defined once they are complete. They work with test engineers in the Agile Teams to identify, create, and execute the right set of tests. Any defects arising from these tests are triaged, residual risk assessed, open items prioritized, and added to appropriate backlog queues. (Scaled Agile Framework 2021c)

System Engineers use the knowledge of the System Requirements Definition process, which transforms work items at a capability or feature level into smaller pieces that specify the operational needs of an end user. They apply the practices from this process and the Implementation Process iteratively to system elements impacted by feature implementation to generate a techno-functional view that can easily turn into user stories. Since systems engineers are knowledgeable about system behavior and associated complexity in implementation, they can assist the team in arriving at complexity sizing for features and stories — enabling optimal scheduling of iterations. Systems Engineers are also able to objectively analyze test results based on their understanding of the system and hence be judicious in deciding to "accept" a user story. They can make decisions in assigning priority to residual defects and can help the Agile teams plan to address defects along with the implementation of functionality.

ENABLING TECHNICAL INFRASTRUCTURE:

Product Owners are responsible for understanding the amount of effort and the scope of work to build basic technological infrastructure that enables the implementation of functionality. Product Owners need not necessarily provide technical guidance in developing these capabilities but can prioritize and schedule their delivery so that they are available for Agile Teams in time. This requires them to work with system and solution architects to understand the complexity involved and the effort required in delivering technical infrastructure to ensure that adequate capacity is allocated for Agile Team members to accomplish technical infrastructure delivery.

In addition to functional and enabler work items, Product Owners are responsible for working with Product Management and other key stakeholders to help identify and understand Non-Functional Requirements and decompose them into smaller work items that Agile Teams can work upon. Non-Functional Requirements are critical for meeting stakeholder needs and ensuring that systems work according to their intended behavior.

Systems Engineers, as part of applying the Architecture Definition Process, can define multiple system architecture alternatives and enable the selection of one or more solution architectures that meet system requirements and stakeholder needs. They are aware of technical expectations from the solution architecture and can clearly capture the requirements for implementing the technical infrastructure needed to support the selected solution architecture. This makes it effective for Systems Engineers to work with Agile teams to help them prioritize and implement enabler work items.

The Systems Requirements Definition process empowers Systems Engineers with a set of tools and process steps to identify non-functional and performance requirements, and the constituent parts of the System Specification that emerges as the output of this process can iteratively feed into the program backlog at appropriate intervals — especially during Agile ceremonies like Program Increment Planning.

GATHERING FEEDBACK AND CONTINUOUS IMPROVEMENT:

Product Owners are a key part of periodic capability demonstrations carried out by development teams to key stakeholders. They receive firsthand feedback from stakeholders that expect them to process this and ensure that all items that need to be acted upon and would result in a change of system behavior gets into the backlog funnel and are elaborated and prioritized.

Since they are also an integral part of Agile Teams themselves, they take part in an Agile ceremony called the Iteration Retrospective. The objective of this ceremony is to review and retrospect the completed iteration and find improvement opportunities. The improvement opportunities could be based on actual metrics like work items delivered, defects introduced and addressed, and others, or could be based on practices, processes, challenges, and such. Product Owners can contribute to identifying issues and shaping solutions for improvement.

As part of the Project Assessment and Control Process that *Systems Engineers* are well versed in, they have the right set of tools to assess feedback coming in periodically from stakeholders and Agile Teams. These assessments also monitor progress made on the technical front. When this process applies to feedback emerging from retrospective ceremonies, potential corrective and preventive actions can be identified to ensure that the program can meet the schedule and budget constraints and improve the overall efficiency of processes.

Systems Engineers also review designs, specifications, at times, source code, and other artefacts periodically and can provide technical feedback to Agile Teams and steer them towards course correction if required during implementation. This is an added benefit as regular Product Owners are generally non-technical and they will need to rely on other technical staff to accomplish this.

CONCLUSION

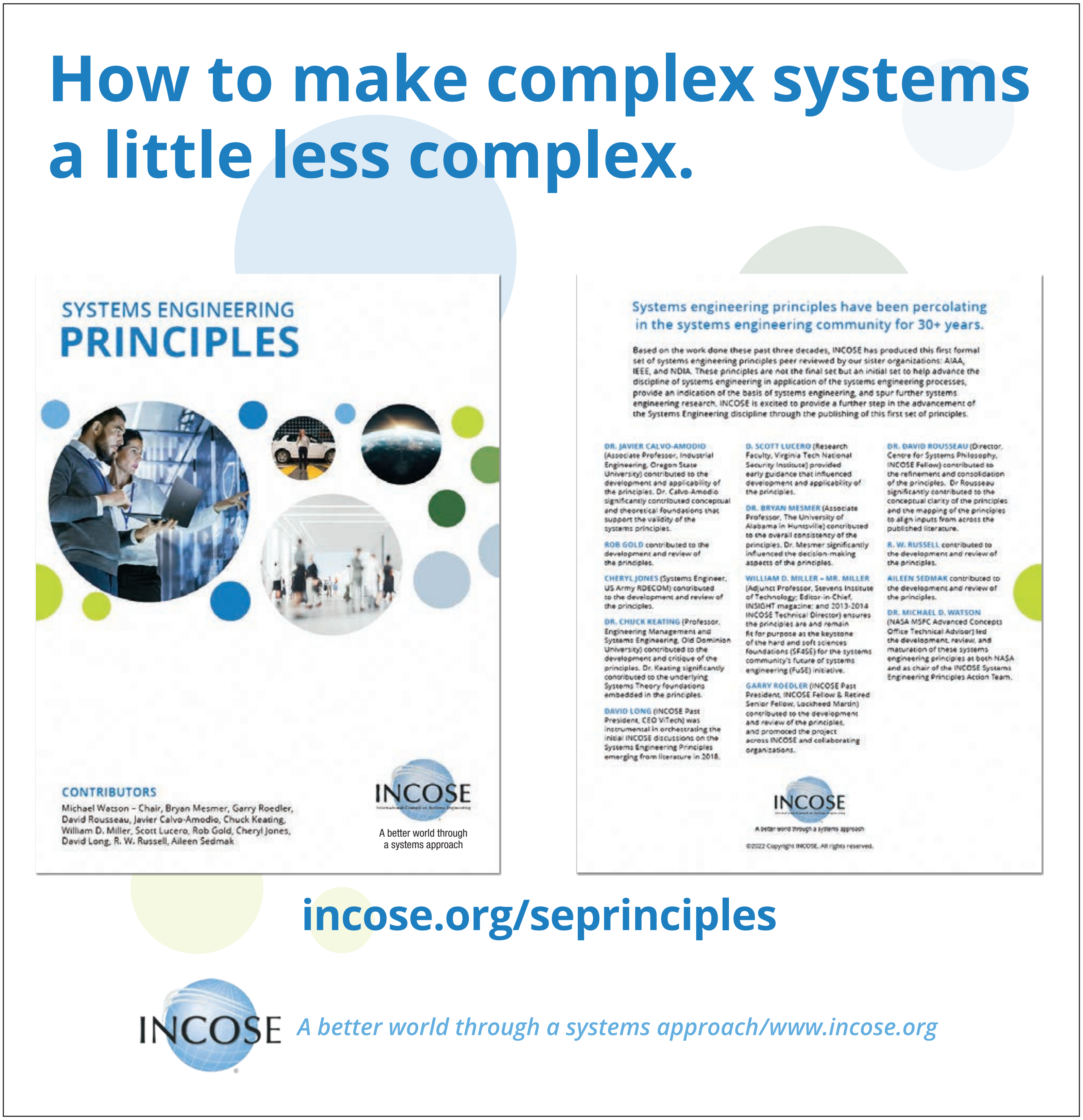
In conclusion, Systems Engineers enabled with Systems Thinking behavior, tools, practices, and skills developed by applying Systems Engineering processes make them efficient and well-suited to play the additional role of Product Owners in the "Agile" world. Organizations and programs would benefit from this approach — both in terms of overall quality and execution efficiency. ■

REFERENCES:

- INCOSE. 2015. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. Version 4. Revised by D. Walden, G. Roedler, K. Forsberg, R. D. Hamelin, and T. M. Shortell. San Diego, US-CA: INCOSE.
- Scaled Agile Framework. 2021a. "Product Management." https://www.scaledagileframework.com/product-management/.
- ———. 2021b. "Product Owner." https://www.scaledagileframework.com/product-owner/.
- ———. 2021c. "Program and Solution Backlogs." https://www.scaledagileframework.com/program-and-solution-backlogs/.

ABOUT THE AUTHOR

Aswin Sukumaran Nair is a Principal Engineer with Carrier Corporation, based out of Hyderabad, India. He is a Systems Engineering Practitioner with over 17 years of experience in software and systems development. He is a certified ASEP and has completed PO/PM certification from Scaled Agile Framework.



Why Mountain Bike Trails Try to Scare You Off

Courtney Wright, courtney.wright@v1decisions.com

ABSTRACT
INCOSE Certification looks for multiple pillars of skills. Some candidates have a pillar in Systems Engineering, a pillar in other Science, Technology, Engineering, and Math (STEM) skills, and a pillar in a domain (or multiple pillars in these areas). What does that look like on an application for INCOSE Certified Systems Engineering Professional (CSEP)? How do our Certification Application Reviewers (CARs) assess it? This article goes far from the commonly discussed domains and products of systems engineering and looks at how leading a bike ride can be done better by using systems engineering principles. It concludes with whether applying Systems Engineering in such an unusual case is enough to qualify a candidate for CSEP.

WHAT CAN GO IN THE Π?

If systems engineers have Π-shaped experience, with one area of depth in systems engineering and the second area of depth in another engineering discipline or a domain, what are the bounds of those other areas of depth? What if a person has depth in systems engineering and civil engineering, and they apply those to building LEGO sculptures? What if their depth is in systems engineering and psychology, and they work in the domain of artificial intelligence for self-driving vehicles? It is challenging for the INCOSE Certification Program to determine what background and experience count as systems engineering and what does not. In particular, the Certification Application Reviewers (CARs) grade on a pass/fail scale, so they must know the edge cases between systems engineering and not-systems engineering.

To define these edge cases, let's look at a hobby: mountain biking (MTB). Not building mountain bikes or designing trail features but rather preparing for and executing rides. This is an interesting case because the system of interest is an experience, not hardware, software, or process. Could we also produce systems engineering products related to this experience? Yes. The National Interscholastic Cycling Association (NICA), which oversees the youth mountain bike leagues in the US, provides risk management training and requires the development of risk mitigation plans. Coaches also continuously solicit requirements from stakeholders. They employ systems engineering concepts in their guidance and coaching, even though they don't use standard systems engineering terminology.

Are systems engineers better mountain bikers than those without systems engineering competence? Maybe not, but they're probably better mountain bike ride leaders. Do all mountain bikers apply systems engineering? No. Sure, it is possible to go mountain biking without using principles from systems engineering, but it probably won't turn out as well.

The unique abilities of the systems engineer are useful for improving the mountain biking experience. Risk management, lifecycle considerations, and system requirements are all relevant to the systems engineer who is going out into the woods with a bike. Do we always have a plan? No, but we have an intentional decision about whether a plan is needed.

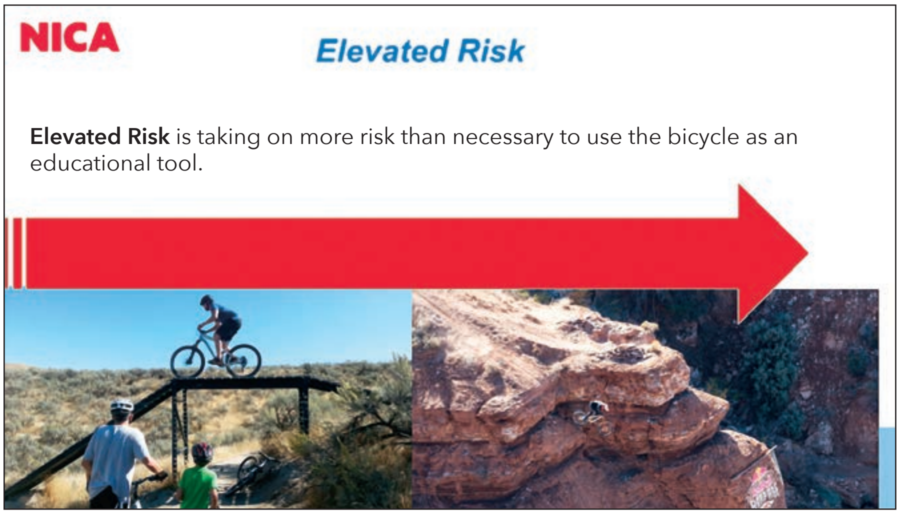
Within a youth mountain bike team, coaches teach systems engineering to their student-athletes, their parents, and their assistant coaches. They don't call it systems engineering. They call it planning and risk management.

BIKE CHECKS AS DECISION POINTS

The first thing we do at every bike team practice is check our alphabet. A is for Air in the tires; B is for Brakes working properly; C is for a clean Chain; D is for a smooth-shifting derailleur. Then, we gently lift and drop the bike and let it bounce on its wheels to listen for E: Everything Else that rattles or clicks or falls off. We don't write down requirements for a working bike, but this activity looks a lot like verification that the working bike requirements are being met.

Our primary verification method is inspection. How else do we verify? Lacking the formality of coaches (independent verification), most athletes do a loop around the parking lot on their bikes before they hit the trails. They don't have a list in mind of what they're looking for. They're mostly giving themselves time to remember what they left in the car before they get too far away. (Water bottle, helmet, and gloves are common things to forget.) But they're also gathering information about the ABCs listed above, along with ensuring their seat is at the right height and position.

Why don't we have our student-athletes verify through demonstration instead of



inspection? Efficiency and confidence. It is very common for our student-athletes to have low air pressure and forget their water bottles. We want to figure this out quickly, so we can fix them all simultaneously (we have multiple pumps) with minimal delay to the other student-athletes. We want to be confident that the inspection occurred. With many students performing demonstrations at the same time, we lack confidence that they (or the coaches) will check and notice problems, and we also have an increased likelihood that the problems are noticed at staggered times and addressed inefficiently.

The other common ways of verifying a bike's functionality for a trail ride are to test it by going through a quick set of actions: going up a curb, going down a curb, turning tightly, and stopping quickly. Riders may take these actions on their own, or a trail may force riders to perform these tests by positioning several challenging technical features near the beginning of the trail. These challenges at the entrance to a trail are called "qualifiers," and they are intentional to help riders test their fit with a trail before they get too far in. This trail design is a beautiful solution to many problems. It keeps unsafe bikes (or riders) off the deeper parts of the trail and if someone injures themselves on these early features, at least that injury is in an area that is easier to reach by emergency personnel. Of note, the most challenging features are placed deeper in the trail to increase the likelihood that only skilled people attempt them. The early-trail features are designed to scare off, not injure.

What to do if a bike fails the verification activities, either its bike check or the early features on a trail? Like a good systems engineer, the rider will then consider whether to modify the ride requirements or to modify the solution chosen to meet those requirements. If the requirements for the experience are to ride with certain people, at a certain location, for a certain time, and at a certain difficulty level, could one of those be decremented while still meeting the other requirements? Mountain bikers typically describe their ride plan in terms of the solution, but the requirements are typically more like, "Hang with my friends for a couple of hours, getting exercise and scaring ourselves a little bit." If the tires won't hold air, a rider may choose to bring a pump and take a break every thirty minutes to refill, patch the tire, or do a different activity with friends. If the trail features are too challenging, riders will typically choose to take a different trail (made easier if they have chosen a park with varied difficulty levels) or to walk past that feature.

FUEL AS A LIMITED RESOURCE

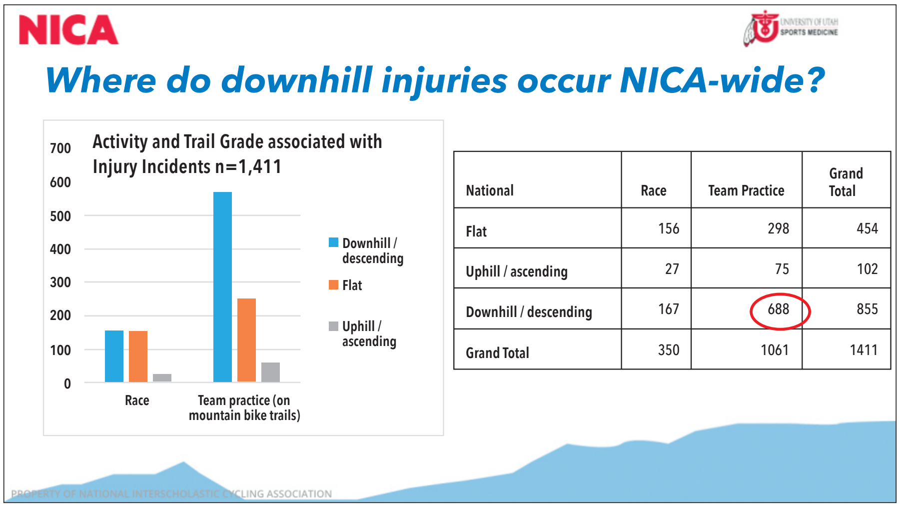
Riders must fuel themselves before the ride and bring in fuel for during the ride. Like with bike checks, if a rider is not properly fueled, they might choose to modify their plans for the day. It is common for riders to realize they are insufficiently fueled only after they are far from the trailhead where the food and drink are stationed. Thus, smart riders bring extra fuel with them. When consequences of under-fueling are highest, riders plan more carefully.

Some situations with higher consequences are bike races, isolation, and extreme weather. When planning for a race or in severe weather, riders will want to minimize events that slow them down because even a small delay can be significant to how someone finishes in the race or survives outdoors. When riding in a group in an isolated area or by oneself anywhere, a rider won't get extra help quickly. In all cases, the systems engineer ride planner will pack and ride differently depending on these factors. As part of youth mountain bike coaching, coaches are required to carry a basic first aid kit (and have current first aid and CPR training) for all practices and rides. They are also required to have an emergency action plan for every location where they ride, including knowing how to evacuate if there is an injury. A systems engineer serving will similarly decide what they pack based on the ride plan.

RIDER SKILL, TRAIL RATINGS, AND SYSTEMS ENGINEERING DECISION-MAKING

There are varying skill levels among mountain bike riders. Bike trails are marked according to how difficult they are using the green (easy), blue (intermediate), and black (advanced) rating scale. These skills are one potential area of competence for a rider. These skills are comparable to those of a technician, writer, or coder — they are purely technical skills that can be independent of leadership and managerial skills.

Some riders are excellent jumpers, have long endurance, or can get through narrow passageways. Other riders may have none of those skills but are fast or confident. A perennial challenge of riding in a group



National	Race	Team Practice	Grand Total
Flat	156	298	454
Uphill / ascending	27	75	102
Downhill / descending	167	688	855
Grand Total	350	1061	1411

is deciding who will be in the front of the group. If the group wishes to stay together, this person will make many decisions that affect those behind them. If a group does not decide in advance who will be at the front of the pack (also called the ride leader), this decision is made informally by who chooses to start riding first. This confident person is not necessarily the best leader for the group. Riders rarely discuss their objectives and risk tolerance before starting a ride.

A ride leader's decisions include which trails to take and the maximum speed at which to travel. The ride leader also decides when to take breaks and when to restart. Using the INCOSE UK Competency Framework's (INCOSE UK 2010) four levels of competency, a ride leader will need at least "Practitioner" level competence in most systems engineering areas. Someone with lower systems engineering competence may be able to lead rides on familiar trails based on their experience on those trails or moderately familiar trails if they have observed a highly skilled ride leader. I'll emphasize again that high skill in ride leadership is not the same as high skill in riding, though they may overlap.

A ride leader with high skills customizes the ride according to the participants, their objectives, and the environment, along with other considerations. A ride leader with moderate skills may be able to lead an individual rider with one of a few, narrow objectives. This person might "tow in" an inexperienced rider, modeling a line and announcing when to switch gears. Through experience or training, the ride leader may develop expertise in the role. Certifications for mountain bike instructors focus on relationships and communication, not on technical bike skills.

MONITORING AND CONTROL

Like systems engineering, MTB has tools that make ride planning and execution more likely to achieve its objectives. Tools are accessed either through smartphones or through dedicated bike computers, which are typically mounted on the handlebar of a bike. Two example tools are TrailForks and Strava.

TrailForks is useful for both ride planning and capturing/distributing lessons learned. It includes maps, photos, and reviews of trails, and it is searchable. Strava tracks ride paths, start time, end time, and speed. It is typically not used during a ride but is used for tracking individual rider performances over time. Both tools display real-time GPS positioning on a map.

Ride planners commonly use weather apps and social media to get recent information. All of these tools are used for planning rides, and the mapping tools, along with other GPS mapping apps, can be useful for risk mitigation during a ride. The bike-specific tools are further useful in their tracking of technical measures.

Did someone say technical measures?! Gathering information to make decisions and take action? If a ride leader were setting technical measures, they might be:

- Ride distance
- Ride duration
- Average speed
- Maximum speed
- Fastest speed of other riders on the same route

These measures may be listed for an entire ride or segments within it. They can be used for planning future rides with the same riders or with others. If a ride was slower than expected or the duration was longer than typical, the ride planner may take note of that to account for it in the group's next ride. They may do alternate post-ride activities, like more food, medicine, or stretching than usual.

MTB ON A CSEP APPLICATION FORM

Would a Certified Systems Engineering Professional (CSEP) applicant be able to claim their work experience as an MTB coach as systems engineering experience? What about just leading rides among friends? One of the obstacles is the amount of time the work would have to be performed to be relevant. If someone is a full-time nature guide and they are incorporating systems engineering in their way of approaching their work, they could claim it in a SEP application. If they are a nature guide or bike ride leader for two hours each week, it would take nearly two years for them to accumulate "one month" of systems engineering experience. That experience would have to be divided across the multiple systems engineering experience areas they were claiming. A SEP applicant who had coached or led mountain bike rides for ten years might find it worthwhile to document that, either in one experience area or divided across multiple areas.

Some ways a CSEP applicant might claim systems engineering experience related to mountain biking are:

- Requirements Engineering — working with stakeholders to determine their objectives for a particular ride or series of rides; their objectives are rarely expressed clearly as requirements
- System and Decision Analysis — using sound and repeatable judgment to make decisions about which trails to take given the variables on a particular ride, including trail conditions,
- Systems Integration — bringing together multiple riders and their objectives can mean putting them in a particular order within a ride or sending some of them on different paths [A common way to integrate varying speeds is to have the fast group repeat (called "sessioning") a section of trail while the slower group does it just once, resulting in everyone riding for about the same amount of time on the same trails but the faster folks traveling a longer total distance. This is particularly useful because it allows water breaks to line up in duration and location, which brings the social aspect to a group ride that is an objective of most participants.]
- Verification and Validation — checking that bikes are ready for a ride and checking that riders are ready for a particular trail, including both design of those checks and assessment of the results
- System Operation and Maintenance — leading a ride and dealing with the common obstacles that arrive, such as water bottles falling off bikes, minor falls from riders, group members

getting lost, and branches or unrideable sections in trails

- Technical Monitoring and Control — tracking heart rate, speed, and water consumption of riders to see how their bodies are doing compared to expectations, then adjusting the remainder of the ride to get closer to objectives. For example, I had a rider who went through their water faster than expected, so I sent them back for a refill and had them meet back with the group after a challenging section
- Information and Configuration Management — storing and accessing previous routes and notes about them
- Risk and Opportunity Management— developing both a risk management plan, including what options are acceptable and by whom, and risk mitigation plans for identified risks; remaining open to opportunities to try new lines (paths) through trails
- Lifecycle Process Definition and Management — recognizing and adjusting to the changes to riders, their equipment, and trails that come naturally with time

Excellence in any discipline is achieved through intentional action. The systems engineering concept of tailoring comes into play and may disguise some of the systems engineering being done in support of mountain biking and other activities. Risks may have been identified and accepted, for instance, and that is still a reasonable path through the risk management process.

Going back to our CAR, the task of reviewing an application is easier than the task of reviewing a body of experience. The CAR has a limited set of information to assess and cannot extrapolate or consider examples not documented. The CAR also has the benefit of reviewing the mapping the applicant has created. Although the applicant may have gotten help in the mapping, it is a supporting indicator of their competence if they submit a mapping that makes sense with their written descriptions.

Could someone be approved as a CSEP based purely on their work as a mountain bike ride leader? Definitely. However, at a slow pace of six miles per hour, common for a combined session of teaching and riding, they would still have to ride at least 60,000 miles to get their five years of systems engineering experience! ■

REFERENCES:

- INCOSE UK. 2010. "Systems Engineering Competencies Framework Issue 3." https://incoseuk.org/Documents/Groups/UKChapter/SE_Competencies_Framework_Issue_3.pdf.

ABOUT THE AUTHOR

Courtney Wright is an INCOSE Certified Systems Engineering Professional and the Associate Director of Certification for INCOSE. She has over twenty years of experience in systems engineering and approximately two years of experience in mountain biking. She enjoys applying the principles of systems thinking and solving challenging problems, both indoors and out.

SEPTEMBER 2022
VOLUME 25/ ISSUE 3

Teaching Systems Engineering Practices Using Principles from Studio Art Education

Tom McDermott, tmcdermo@stevens.edu; and **Molly Nadolski,** mnadolski@agnesscott.edu;

ABSTRACT
We have created a cross-disciplinary learning program across studio art, systems thinking, and systems architecture to improve systems competencies. Previous research produced a framework for relating learning concepts across these three domains. We describe how to apply these concepts to any learning situation to increase systems competencies using systems thinking and systems engineering tools. We have applied the framework in educational settings to teach systems thinking, critical thinking, and systems engineering using a tailored classroom approach derived from studio art classes. The studio format encourages the students to build their skills across a portfolio of work focused on creating appropriate views of a system, communicating them, and gaining critical review – three core aspects of the systems thinking process. An effective systems engineer conceptualizes and solves problems, addresses stakeholder concerns, explores and composes solutions, and manages system evolution. The use of art, as embodied in systems-related tools, has proven to be an effective method for developing these skills. This article presents the framework and an example graduate level Systems and Critical Thinking class taught using the studio art approach.

INTRODUCTION

In our work, we have found that the creative arts can serve to link skills and learning methods across the disciplines of systems thinking and systems architecture, ultimately improving systems engineering competencies. The artist, the architect, the systems thinker, and systems engineers use visual and textual forms to conceptualize the system and compose meaningful solutions. We found that there are a set of definable learning concepts linking art, architecture, and systems thinking that serve to improve the conceptual and compositional skills of a systems engineer at any learner level.

The INCOSE revised definition of a system recognizes the relationship between conceptual systems and physical systems as representing "meaning" versus "behaviors." The definition further states, "Conceptual systems are abstract systems of pure information and do not directly exhibit behavior, but exhibit '**meaning**'" (INCOSE 2019). Most modern systems pass through multiple stages of meaning, first to the development and investment stakeholders (concept definition), then to the various users (realization and use). Great artists, systems thinkers, and system architects excel at conceptualizing and composing meaningful representations of the system to manage complexity and create stakeholder understanding. The presentation of information in any context or life cycle stage must be based on a judgement about what is important to include or not. Observers derive meaning from their relationship to that information. Among other things, effective systems engineers develop the unique talent to abstract information and communicate it to stakeholders in meaningful ways.

Architects use artistic practices to manage the complexity of a system as a means of communication between the their work and the client. Similarly, systems and critical thinkers use these practices to manage the complexity of a **situation** across the broader system context and all affected stakeholders. The use of art represents a bridge between a stakeholder and a complex system or situation that helps to improve stakeholder understanding and guide their activities. We want to use this bridge to accelerate the mastery of systems-related methods and tools.

EDUCATIONAL APPROACH

Our educational approach uses a **studio art** setting, which encourages effective communication and critique across portfolios of work, to improve systems-related skills. Systems engineering concepts are relatively easy to teach but more difficult to put into practice. A similar problem exists in art education, but art educators design their courses around foundational learning and repeated studio-based iteration to balance learning and practice. Many systems

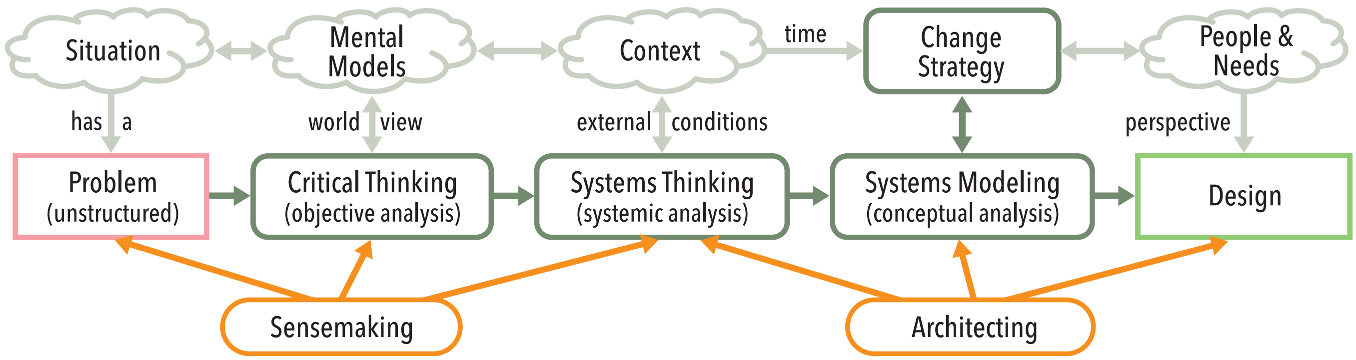


Figure 1. Systems and critical thinking process flow

concepts like abstraction, simplification, decomposition, and synthesis are more like art, requiring iterative practice with feedback to gain mastery (Salado, McDermott, Davis, and Moral 2019). We teach systems skills, methods, and tools in a studio art setting to create an environment for iterative practice. Creating linkages between art (communication and aesthetics), systems thinking (conceptualization), and systems architecting (composition) makes the development of good systems practice more intuitive to the learner and thus easier to master.

Virginia Tech was the first to apply the methods described in this work to a graduate course on Systems Engineering Strategies (Salado, McDermott, Davis, and Moral 2019). Feedback from that class indicated the students felt the class activities were closer to a real work environment and were more engaging to learn. In addition, the students improved their ability to address a problem from multiple perspectives, a core systems thinking ability.

In this article, we discuss a Systems and Critical Thinking class we have taught now for three years in a graduate degree program in Technology Leadership and Management at Agnes Scott College in Atlanta, Georgia, US. This course introduces systems thinking and systems engineering to non-technical career professionals who need to make effective decisions about using technology in organizations. As we matured this approach, we learned we could standardize on a process flow that takes students through a disciplined problem-solving methodology. The process flow leads the student to iterate from situational understanding to systems exploration (sensemaking) to effective solution design (system architecting) using various critical thinking, systems thinking, systems modeling, and design methods/tools. See the process flow in Figure 1.

LEARNING MODEL DEVELOPMENT AND THEORY

The flow of Figure 1 is highly focused on developing good conceptual and compositional skills. Each stage of the process is a quest to accurately communicate the whole of the system as a set of **views**. A view communicates a representation of the whole system with respect to a stakeholder set of perspectives or concerns (ISO 2011). Systems engineers are trained to create these views. However, they are not always led to realize that problem solving is a holistic and disciplined process involving many perspectives and views. In the workplace, they tend to be unguided, requiring a significant amount of time to absorb the practice. Non-systems engineers easily understand the concept of a set of views, but they seldom get the training necessary to create them or to understand them as a problem-solving process flow. Everyone can improve their systems thinking and architecting skills by practicing similar development of these views.

We believe frequent calls for "better systems thinking" in all professional disciplines is a desire for employees to gain a good balance across the conceptual, compositional, analytical, and methodological skill sets. The framework of Figure 2 depicts this balance. We derived it from college advanced placement guides for music education (College Board 2016) and not from existing systems literature. Education in engineering, technology, and business-related disciplines tend to overly focus on analytical and methodological competencies and fall short of the conceptual and compositional. In the arts and other disciplines like physical architecture, students must first master conceptualization and composition before they move to analytical and methodological approaches. Art education encourages conceptualization and composition by using effective communication and critique. It can effectively improve this balance.

Table 1 presents a set of learning outcomes that link the creative arts, systems thinking, and systems architecture. Table 2 presents a set of learning concepts that guide education in system related disciplines, organized into these four knowledge areas (McDermott and Salado 2017). We believe this has had initial qualitative validation as an educational framework based on experience and feedback in our educational approach, but objective validation requires additional use of the framework in similar courses.

Concepts 1-4 in Table 2 represent conceptual approaches, and concepts 5-10 represent compositional approaches. We only cover the analytical and methodological approaches in concepts 11-12 briefly since systems literature already covers those concepts well. However, domain-driven use of sketch, storytelling, mapping or modeling, and public communication are themselves

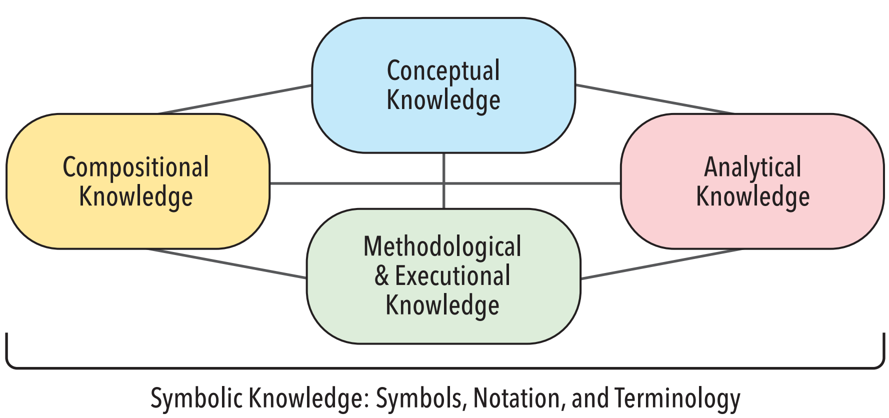


Figure 2. Knowledge framework practice areas (McDermott and Salado 2017)

Table 1. Learning outcomes (McDermott and Salado, 2017)

Learning outcomes	**Art:** Understanding of composition, inventive design, risk-taking, creative expression, breadth of work, *individualized transformation of concepts*	**Systems Thinking:** Holism, composition of mental models, creative thinking, framing complexity, breadth and depth, *group strategies of transformation*	**Systems Architecting:** Composition of operational & technical design, methods & execution, planning, breadth & depth, *group communication of transformation*

Table 2. Learning concepts (McDermott and Salado, 2017)

Learning Concepts	Knowledge
1. Abstraction	Understanding symmetry between your internal context, stakeholders' business context, and the observed system need or problem.
2. Precedence	Finding, interpreting, understanding, and communicating unprecedented concepts.
3. Time	Communicating strategies and attributes of change.
4. Heuristics and Patterns	Learning to see patterns of change and to relate these in narrative and visual forms representing emergence.
5. Decomposition, Recomposition	Using modeling methods that convey aesthetics, purpose, relationships, circumstances, objectives, goals, and strategy.
6. Boundary setting	Observing and framing, boundary-setting, context setting, scenarios.
7. Simplifying	Determining and communicating abstraction levels, lumping, or splitting components.
8. Synthesizing	Composing, interrelating, reconnecting to express emergence. Using different types of views.
9. Focusing	Emphasizing, centering, contrast & balance, perspectives. Using different types of views.
10. Communicating	Aesthetics, patterns, colors & shapes, storytelling & facilitation, documentation.
11. Analytical Competencies	Domain-driven technical and business analysis augmented by sketching, narrative, and structural models and data.
12. Methodological and Executional Competencies	Creativity, use of processes and patterns, experimentation & risk taking, ability to engage & gain audience focus, understanding and agreement.

Table 3. Three different contexts of abstraction (McDermott and Salado, 2017)

Art	Systems Thinking	Systems Architecting
Understanding Symmetry between our **internal context** and observed system	Understanding Symmetry between the **situational context** and observed system	Understanding Symmetry between the **business context** and observed system

important skills to practice and learn. As one reads this table, it becomes apparent that the conceptual and composition skills in concepts 1-10 all support the analytical and methodological skills of the systems person in concepts 11-12.

The framework encourages use of architectural design principles in project-based learning to provide the systems thinking foundations in any domain. The architectural views force the students to take a bottom-up and top-down approach, understand context, explore organizational, technical, and business-related aspects of the system, and work together with dialogue to solve problems. Godfrey, Deakin-Crick, and Huang similarly describe the value of systems architecting in an educational setting as a dynamic learning opportunity that is a generative process not guided by established form. They recommend starting with a contextual problem and then exploring the system as student-led discovery with the instructor as facilitator (Godfrey, Deakin Crick, and Huang 2014). We have adopted this approach and learned that focusing on concepts of **abstraction, composition**, and **meaningful communication** is significant to the discovery process.

FOCUS ON ABSTRACTION

Effective abstraction (learning concept 1) is the center of our teaching methodology. Table 3 identifies how art, systems thinking, and systems architecting separately but synergistically focus on creating meaningful abstractions of complex concepts to communicate them in a way the stakeholder community can interpret them. The other conceptual and compositional learning concepts 2-10 in the framework support the abstraction process.

Artists, systems thinkers, and system architects all use creative abstraction to conceptualize new designs. When combined, the three contexts encourage combined exploration of self, situation, and solutions in the discovery and learning process. In the classroom, we pose an unprecedented system challenge to encourage creative iteration. The use of an unprecedented (to the students) challenge forces the learner-centered discovery process.

FOCUS ON COMPOSITION

Compositional strategies support the expression of the system as both a problem to be solved and potential solution opportunities. Problem-solving techniques blend conceptual and compositional tools to build insight and prevent "jumping to solutions." Artists use the canvas, plot, stage, or music staff to describe their composition. The systems thinker uses narrative and models. Architects use views.

Models that are meaningful to an observer require aesthetic design. Exploiting relationships between art forms and systems thinking tools helps the learner to master the ability to communicate aesthetically and with meaningful abstractions. Finding the right level of detail, selectively lumping together or splitting aspects of the system, and conveying these in a pleasing form links the presentation of the system to an observer. Additionally, effective compositional skills communicate the system back to the designers themselves, helping them to creatively explore alternatives that might produce more holistic designs. We want students to produce views that are aesthetically pleasing to the observer, effectively convey meaning, and express the right depth of information.

FOCUS ON COMMUNICATION WITH MEANING

In this work, we are particularly interested in the relationship between a presenter (student) and observers (peers and instructors), and the presenter's ability to create a shared meaning of a complex system. In art, systems thinking, and architecture this is a process of consciously selecting information to be presented (narrative and media) to create the right balance of concrete versus abstract. With respect to views, this requires a balance between freeform and structured representations of information (Whitla 2020). Students and observers require multiple types of views to improve their understanding.

In a classroom setting, we ask students to focus on both current and future states inherent in the central question "how will automation change my job…," present their analysis using "the right amount of detail," and then critique them in peer review on how meaningful their presentation was to the observer (instructors and peers). This is quite like the form of a studio art class. We designed different class exercises using combinations of more freeform versus more structured views to encourage discussion of the student project at differing abstraction levels.

Judgements of a particular student's projects using terms like "exhibits meaning" would be useless if their expression of their resultant system purpose and design ideas was unguided and lacked the diversity in views necessary to describe the concept of a system. The process of creative discovery and diversity of expression that the students experience weekly supports these judgements. Unfortunately, too much of today's student learning discourages creativity and diversity.

The process flow of Figure 1 defines the progression of views. From studio art, the portfolio of **creative** work and progression of views across the students' development provides evidence that supports the assessment of student progress. From systems thinking, the use of **diversity** in types of views and related descriptive narratives build up a full description of the system that instructors evaluate during grading. Development of a portfolio of views using multiple tools across a single system exploration becomes the art studio approach.

LEARNING DESIGN AND EXECUTION

Our student learning design starts with a "central question of interest" that drives a learner-centered inquiry process. We have the students define their own project while centered on a common **situation**. The current situation explored in the class is "*How will automation affect my job (or someone's that I know)?*" This type of question requires the student to address both aspects of the system and aspects of self. Placing the question into a future system instead of a historical example creates the unprecedented challenge that guides exploration and learning. The presented situation centers on the emergence of automation as it relates to their own professional field or one that interests them.

Weekly class assignments follow the process flow of Figure 1 using diverse visual and narrative forms that are critiqued and peer-reviewed by the instructors and peer students. The process flow requires the students repeatedly "zoom out, then in" on their topic to build holistic insight. The art studio setting encourages developing a peer-reviewed student portfolio of system artifacts across their system of interest. Each section of the course explores using different systems thinking and systems engineering tools. This requires the students to develop multiple views of the situation as the class proceeds. Most of the students do not have in-depth knowledge of the technical aspects of automation at the personal or disciplinary level. Much of the work and research done throughout the semester for them is novel and undiscovered, leading to an emphasis on creative exploration. By the end of the course, most have developed

Purpose: By creating automated digital document processing systems, staff and faculty at colleges and universities **can increase focus on student learning and development and other tasks that cannot be as easily automated**. The mission points of this system include:

a. Minimizing the amount of time spent on paperwork and administrative tasks
b. Minimize costly errors
c. Increase student satisfaction and retention

Figure 3. Example student cartoon and system purpose derivation (Prince 2020)

sufficient depth in their selected automation technologies to lead a system development process — they have become systems engineers.

EXPLORATION 1 – SYSTEM PURPOSE AND PERSPECTIVES

We start by having the students refine their "central question of interest" relevant to their domain. The example discussed in the remainder of this article are from an individual working as an academic advisor domain who explored "*how can automation of administrative tasks increase my ability to focus on student learning and development?*" We obtained permission to use these artifacts.

A "**central question**" is a systems thinking tool that forces an initial activity to define and bound the problem space, as opposed to having a broad subject area to study and explore. The student then must define the system more specifically and describe its **purpose** and related **missions**. We have the students capture system purpose and missions as a simple 4-panel cartoon starting with "what am I observing" (panel 1) and ending in "what is my desired outcome" (panel 4). This is the first view they must produce. In Figure 3, we present an example from the selected student's project. We find that the cartoon exercise helps the learner move quickly through the "fuzzy front end" of the system design process by forcing consideration of purpose, function, and form. The cartoon also requires that they consider precedence and sequencing of the current state to emergent system outcomes. Finally, it creates comfort with the use of art to learn in the class (we are assessing meaning from their art, not the quality of their art skills).

The students then research their actual "system" at play and select several articles that address the area of interest using different author perspectives or views. This is the core of the critical thinking process. The students must establish a core research base to ensure their system exploration is informed, accurate, and systematic, and to ensure diversity in **stakeholder needs and perspectives**. The students use a rich picture to communicate their research (Checkland 1981)(Checkland and Scholes 1999). By exploring diversity in perspectives, then linking them in a rich picture, they must consider and capture the areas of **relationships** between them.

EXPLORATION 2 – STRUCTURED NARRATIVE

The first two assignments focus on capturing others' perspectives on their project ideas. From here, we focus on the details of the system and get the students to begin synthesizing their system to explore it further. We start by having the students move from the visually rich picture formats to the written format, using storytelling and structured narrative. Narratives provide written documentation of the problem and potential solution space and are a method for not only organizing and collecting their data but communicating it as well. Further, narratives are necessary to explore system **emergence**. As their research becomes more comprehensive, we ask the students to place the system and data in a story that would describe the outcome of the solution possibilities. This story is a creative writing assignment, encouraging that they name their system, describe characters who develop and use the system, and remember to express the purpose of the system. See the sample shown in Figure 4.

Structured Context Narrative – **Connect the Dots with SCHEDULR**

Legend
Actors
Processes/Resources
Domain/Circumstance
Structure
Future/Opportunity

Spring semester on a college campus – it's a season bursting with possibilities as winter gives way to warmer temperatures, seniors finish their final classes before graduation and the incoming first year class for the next year is solidified. For academic advising offices and faculty advisors there is a flurry of activity as current students try to figure out what classes they are going to take. Every year, there is much stress as academic advisors and the registrar's office work to manually schedule courses based on faculty interest and student needs. Additionally there's always a few students that don't graduate on time because courses are not being offered when they need them. This year, for the first time, at Johnson-Rivera College, students, faculty, and staff are utilizing a new course selection, scheduling, and tracking system that ensures that students are not only staying on track to graduate, but are also having the opportunity to take courses that fulfill their interests and professional goals. This program utilizes artificial intelligence and is known as SCHEDULER – Schools, Colleges, and Higher Education Data Unification and Locator Resource. Ada Torres, an incoming first-year student, will be part of the first group of students to benefit from SCHEDULR throughout their entire college education.

In June before classes begins in August, Ada has a virual meeting with their academic advisor about their goals, interests, and passions. SCHEDULR listens in on the call and can add data points to the student's file based on the information discussed in the meeting. By automating the process of note taking and filing for these conversations, advisors have the

Figure 4. Example student structured narrative excerpt with color coding (Prince 2020)

We also ask the students to organize the narrative to highlight the structure and behavior (processes/resources) of the system, stakeholders (actors), context (domain/ circumstance), and future opportunity space. After the students present their stories, we ask them to iterate their stories from the critical feedback, and to color code the narrative with highlighting key words. The color coding identifies the language and ontology needed for them to begin to build the formal conceptual model of their system. In the "studio" the students read their stories aloud, then hold open discussion time related to critical review of the quality and meaning of the narrative.

At the midpoint of the class, the instructors lead a **facilitation** session on how their systems might emerge across the "automation" situation space. The purpose of this session is to both introduce the students to a systems thinking facilitation process "in action" and to have them focus further on how their emergent system will look. The joint facilitation results also help the students see the common themes and outcomes in their projects as well as unique aspects. In the learning design, this exercise helps to build anticipatory skills associated with **precedence** and **time** in the framework. The students then create and present a second structured narrative that describes emerging themes for their project.

EXPLORATION 3 – VISUAL MAPPING

Visual mapping is the core of the conceptualization and composition process. To introduce them to systems mapping and diagramming, we use a systemigram (Boardman and Sauser 2013) followed by a more formal causal loop diagram from system dynamics (Forrester 1971). The process of developing these maps requires them to diagram the emergence in the system using their written narratives, which they can do from the color coding and other highlighting of text. This creates a visual representation of their system that complements the written version and captures the resulting change in a form that visualizes relationships and flow. Figure 5 shows an example of the student's systemigram.

The mapping "rules" formalized in the systemigram and system dynamics models are related to **abstraction** and **meaning** – how to organize the links, nodes, and map components thoughtfully and methodically to tell their story using a diagram-based **conceptual model**. The

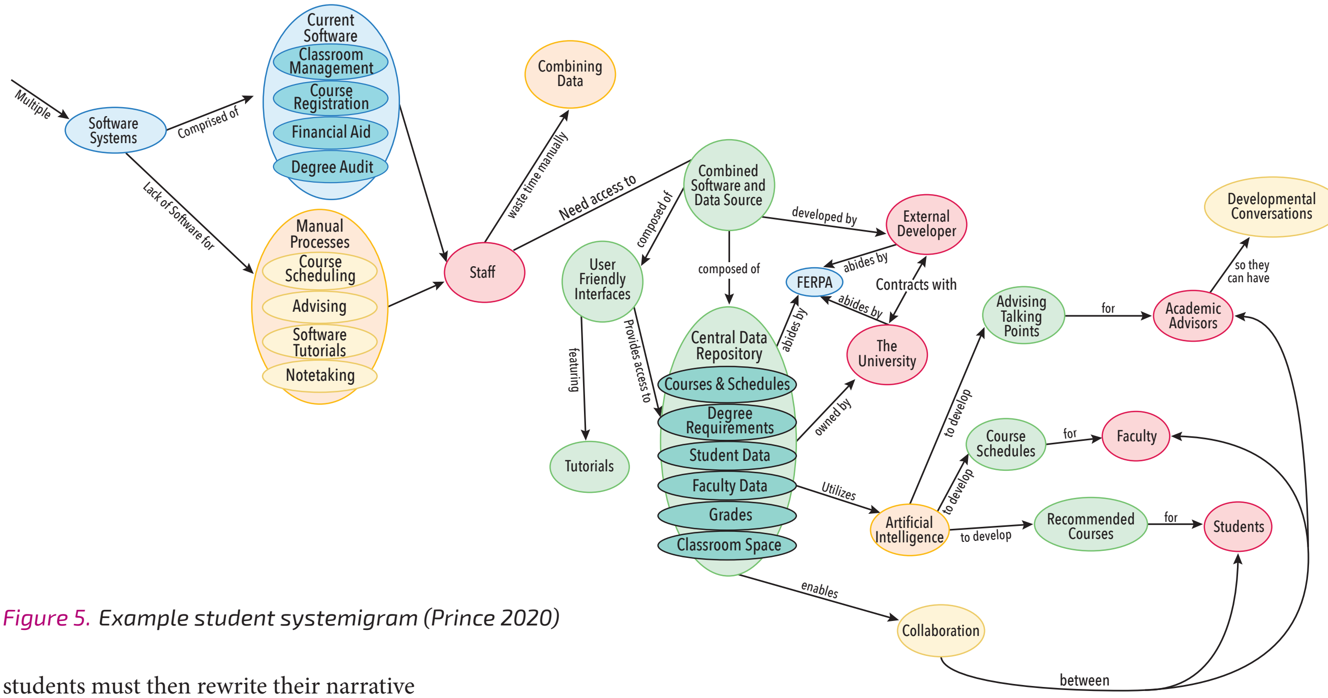


Figure 5. Example student systemigram (Prince 2020)

students must then rewrite their narrative a third time to match the diagram, creating both a narrative and modeled "story."

At this point the students understand and can express their system from many different perspectives. They are ready to apply their learning to date to an effective development strategy.

EXPLORATION 4 – REALIZATION STRATEGY

Once a solid mapping of the system and context is complete, we have the students consider how one would intervene in the system to produce the desired changes. In the class we use two tools for this: a values hierarchy (our derivative of an objectives tree) and a logic model (system intervention strategy). During a project or system analysis, you will have to choose between long-term outcomes that affect how you plan to develop and deploy your project, product, and change. At these points you often must also choose between two things that you might value. In our course, we ask the students to explore how their project value will and will not align with their personal values. For some students, their subject areas are inherently value oriented, so it is an easy tool to use for value identification and evaluation. For others, it encouraged them to dig a bit deeper and draw out values and ideas for their systems that they were not aware of or had not considered before. Figure 6 is an example of the Value Hierarchy mapping tool for our student example. The tool reads from bottom to top: the roots of the system problem we are solving; attributes of the solution space; consequences, effects, or outcomes of the solution; strategic objectives; and finally, personal or organizational values.

The statements in and structure of this diagram help the students construct a formal change strategy in the form of a logic model.

The logic model is a popular tool to clarify and depict a program of intervention within a system (W.K. Kellogg 2004). The model aides the students thinking not just in the future, but also identifying the desired long-term outcomes. They also identify the measurable constructs in their models, and further specify the causal relationships among those constructs and the factors that might influence those relationships. See the example student logic model shown in Figure 7. The goal of this exercise is to fully link the system purpose (identified as a goal statement in the figure); the problem to be solved, long-term goals, and shorter-term goals (the three horizons of change); the entities/resources and functions/activities (these are informed by the color-coding in narratives and the systemigram); and the means of measuring success (outputs). The logic model is a final integrating framework for their study.

The last assignment in the class is a final structured narrative that links all the differ-

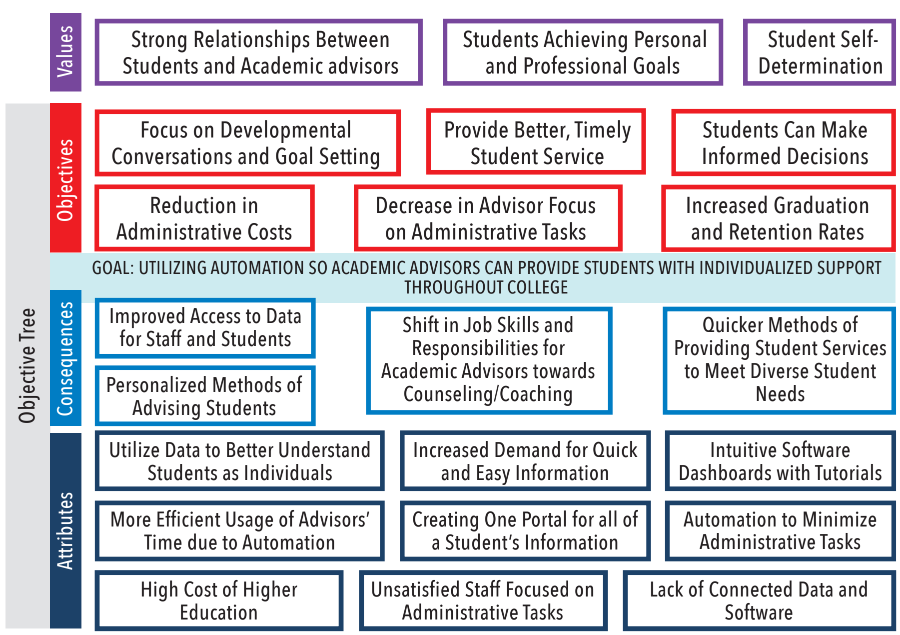


Figure 6. Example value hierarchy (Prince 2020)

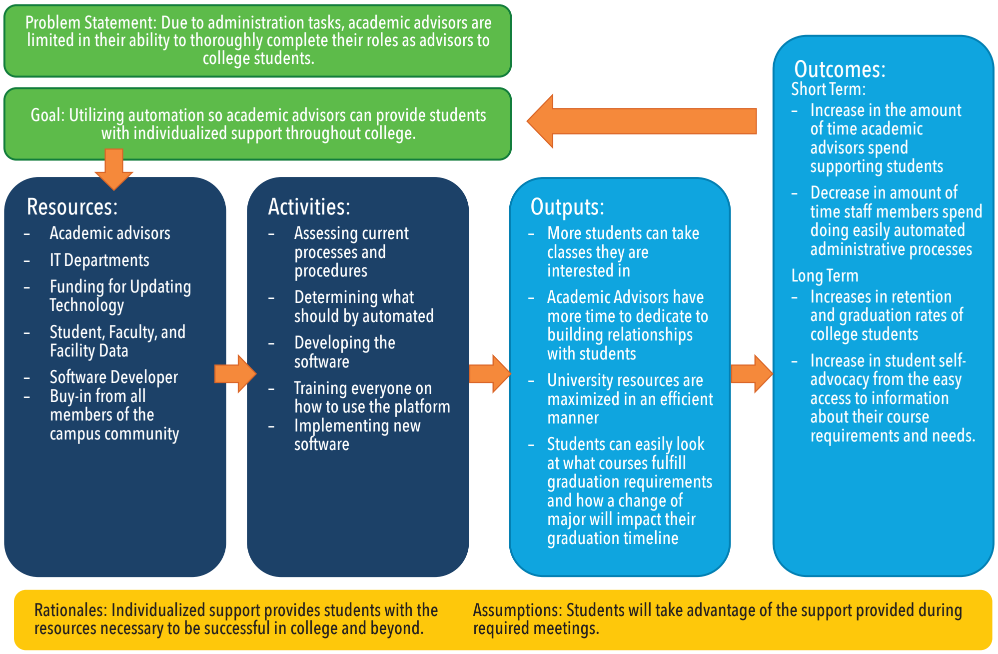


Figure 7. Example student logic model (Prince 2020)

ent tools and created artifacts into a single narrated story. We require this to make sure the students revisit their work across the semester and can see the collection of methods and tools as an integrated flow.

ASSESSMENT METHOD AND RESULTS

The class design presents the learning objectives for the class as a series of subjective evaluations aligned with the common learning concepts of the framework. We derive the primary week to week assessment of the students' work using the art studio format. Our central course effec tiveness question is "can the student represent their emergent automation system design in a way that is meaningful to the observer?" We judge this in the studio format using peer review by other students and the instructors. As with studio art, the instructors in this course are practicing systems thinkers with a history of successful consulting in enterprise domains. The instructors represent the professional review of the students' work. The studio art format requires that the students reflect on their work versus that of other students with each assignment to provide critical feedback. This creates immediate feedback and learning on the methods and tools. This is clearly a subjective evaluation process.

We complete a qualitative evaluation of each student-presented artifact using a series of templated questions that assess whether the artifact was at the right abstraction to convey meaning. Artifacts that are either too "high level" to accurately model the system or "too detailed" to convey its outcomes on the project context produce lower grades. The students almost always improve their ability to abstract as the course proceeds. This leads to our second course effectiveness question, which is "did the student define their system at a sufficient level of detail to realize the design?" This is primarily evaluated by the instructor serving as expert, which is difficult to scale, but could also be done by a subject matter expert review panel.

Finally, the instructor grades for the correct use of the tools for each assignment. The students must correctly and completely use each tool and demonstrate this in their presentations. Students are always given the opportunity to iterate on their assignments if needed to improve their scores. Opportunity for iteration is essential for this to be a realistic systems process.

The student feedback has been very informative in our qualitative evaluation of the framework and related studio learning approach. As with the Virginia Tech class, students genuinely like the creative approach used in the course. Some relevant feedback indicates that the students recognize and appreciate the systems principles learned in the class:

- The systems tools really helped me to narrow down my area of interest which is hard for me.
- With the tools you can really follow along and see the interactions that need to play out in the system.
- System diagramming was key to see the values and outcomes I want as linked to solutions in the project.
- Hearing from everyone else's project you get a better sense of the different ways that we can look at a system.
- The disciplined class flow was challenging but also accessible, so we can feel more successful in our learning steps.
- The peer review format realistically tests my communication abilities and helps me to see aspects of the system I missed.
- I really liked the creativity aspect of this class; every assignment was honestly just fun to do.
- I really loved how everyone was carried along through the studio format and we had the opportunity to understand everyone else's projects.
- I liked the work and way the narratives kept growing and kept getting more and more detailed. At times, this is overwhelming, but in a good way.
- We get new technology all the time, so you want to make sure when we specify and acquire it, it actually does the things that we wanted. This class has helped us learn that whole process so we can do it ourselves, instead of trial and error like we've been doing. I tried the tools with my co-worker, and we could immediately see how the tools would help us better define our technologies we choose.

CONCLUSIONS

We have learned several things from the framework and associated educational design. First, many of the conceptual and compositional tools of systems thinking and systems engineering apply to any professional domain and can be mastered if an opportunity for practice is given. Second, approaching mastery through the concepts of art and art studio methods is very effective at creating the necessary learning and practice. Third, the framework linking art, systems thinking, and system architecture as a set of learning concepts and outcomes has proven to be very useful in the design, of course, learning objectives for the application of systems approaches in any professional domain (including systems engineering). Finally, we believe that systems and critical thinking are very teachable using the right learning concepts and environment (we describe our approach in this paper). If all students were placed in a cornerstone class that used authentic learning environments to practice these systems skills, we might see the global "need for better systems thinking skills" become something we accomplished. ■

ACKNOWLEDGEMENTS

The authors would like to thank all the Systems and Critical Thinking students at Agnes Scott College in Atlanta, Georgia USA who have participated and contributed to our lessons learned. We in particular would like to thank JLP Prince who generously agreed to share the examples. We would also like to thank Dr. Carlee Bishop for supporting the development of this course and Dr. Alejandro Salado for co-developing the framework.

REFERENCES

- Boardman, J., and B. Sauser. 2013. *Systemic Thinking: Building Maps for Worlds of Systems*. Hoboken, US-NJ: Wiley.
- Checkland, P. B. 1981. *Systems Thinking, Systems Practice*. Hoboken, US-NJ: Wiley.
- Checkland P. B., and J. Scholes. 1999. *Soft Systems Methodology in Action*. Hoboken, US-NJ: Wiley.
- College Board. 2016. "AP Music Theory 2016 Scoring Guidelines." secure-media.collegeboard.org/digitalServices/pdf/ap/ap16_music_theory_sg.pdf.
- Davis, K. A., A. Salado, and T. A. McDermott. 2019. "Lessons Learned from a First Attempt to Teach Systems Engineering as a Studio Art Class Paper." Paper presented at the 126th Annual ASEE Conference & Exposition, Tampa, US-FL, 15-19 June.
- Forrester, J. 1971. "Counterintuitive Behavior of Social Systems." *Technology Review*, 4 January. https://link.springer.com/article/10.1007/BF00148991.
- Godfrey, P., R. Deakin Crick, and S. Huang. 2014. "Systems Thinking, Systems Design and Learning Power in Engineering Education." *International Journal of Engineering Education* 30 (1): 112–127. https://research-information.bris.ac.uk/en/publications/systems-thinking-systems-design-and-learning-power-in-engineering
- Iandoli, L., L. Piantedosi, A. Salado, and G. Zollo. 2018. "Elegance as Complexity Reduction in Systems Design." *Complexity* 2018: 1-10. DOI:10.1155/2018/5987249.
- ISO (International Organization for Standardization/IEC (International Electrotechnical Commission)/IEEE (Institute of Electrical and Electronics Engineers) 42010:2011, Systems and software engineering – Architecture description. Geneva, CH: ISO, IEC, IEEE.
- W.K. Kellogg Foundation. 2004. "Logic Model Development Guide." https://wkkf.issuelab.org/resource/logic-model-development-guide.html.
- McDermott, T. A., and A. Salado. 2017. "Improving the Systems Thinking Skills of the Systems Architect via Aesthetic Interpretation of Art." *INCOSE International Symposium* 27 (1): 1340-1354. https://doi.org/10.1002/j.2334-5837.2017.00432.x
- ———. 2018. "Art and Architecture: Effectively Communicating Models of Systems." Paper presented at the 12th Annual IEEE International Systems Conference, Vancouver, CA-BC, 24-26 April.
- ———. 2019. "A perspective on systems thinking, architecting, and art." *Systems Research and Behavioral Science* 36 (5): 648-655. https://doi.org/10.1002/sres.2622
- Prince, JLP. Student example artifacts collected from The Systems and Critical Thinking class conducted January-May 2020. Used by permission.
- Salado, A., L. Iandoli, and G. Zollo. 2016. "Painting Systems: From Art to Systems Architecting." *INCOSE International Symposium* 26 (1): 773-787. https://doi.org/10.1002/j.2334-5837.2016.00192.x
- Salado, A., T. A. McDermott, K. Davis, and A. Moral. 2019. "Why Not Teaching Systems Architecture as a Studio Art Class?" In *Systems Engineering in Context: Proceedings of the 16th Annual Conference on Systems Engineering Research*, edited by S. Adams, P. Beling, J. Lambert, W. Scherer, and C. Fleming, 267-278. New York, US-NY: Springer.
- Salado, A., and C. Salado. 2015. "Systems Engineering Practices Exhibited in the Creation of a Film Original Score." *INCOSE International Symposium* 25 (1): 1147-1158. https://doi.org/10.1002/j.2334-5837.2015.00121.x
- Whitla, S. 2017. "The meaning curve: Where do you need to be?" *Meaning Guide*, 13 November. http://meaning.guide/index.php/2017/11/13/meaning-curve-need/.
- ———. 2020. "Visualising transformation: the pursuit of shared meaning." *Meaning Guide*, 8 June. http://meaning.guide/index.php/2020/06/08/visualising-transformation-the-pursuit-of-shared-meaning/.
- Vitruvius. 1960. *The Ten Books on Architecture*. Translated by M. Morgan. Mineola, US-NY: Dover.

AUTHOR BIOGRAPHIES

Tom McDermott serves as the Deputy Director and Chief Technology Officer of the Systems Engineering Research Center (SERC) at Stevens Institute of Technology in Hoboken, NJ. The SERC is a University Affiliated Research Center sponsored by the Office of the Secretary of Defense for Research and Engineering. With the SERC he develops new research strategies and is leading research on Digital Engineering transformation, education, security, and artificial intelligence applications. Mr. McDermott also teaches system architecture concepts, systems thinking and decision making, and engineering leadership. He consults with several organizations on enterprise modeling for transformational change, and often serves as a systems engineering expert on government major program reviews. He current serves on the INCOSE Board of Directors as Director of Strategic Integration.

Molly Nadolski is currently a Complex Problem-Solving & Enterprise Strategy Consultant for multi-national organizations and a guest lecturer at Agnes Scott College in the Applied Technology Graduate program. Her focus is on systems and critical thinking as an approach to problem solving and a way to address future needs. She is a former Research Associate at Georgia Tech Research Institute working closely with the Sam Nunn School of International Affairs. Her research interests include issues related to national and international security, geopolitics, cyberspace, urban resilience, decision-making, human behavior, and regional development. Her role involves framing and leading policy research, while exploring the applications of systems thinking methods. Prior to her work in Atlanta, she worked at several consultancies in Brussels, Belgium that conducted EU funded projects focused on technology, innovation, and security.

CSER 2023
conference on systems engineering research

20 YEARS

CSER 2023:
Systems Engineering Toward a Smart and Sustainable World

March 16-17, 2023

IMPORTANT DATES

Submissions Due	December 5, 2022
Notification to Authors	January 9, 2023
Final Papers Due	January 23, 2023
SEANET & Tutorials	March 15, 2023
Conference	March 16-17, 2023

INFORMATION

cser.info/cser2023

LOCATION

Stevens Institute of Technology
1 Castle Point Terrace, Hoboken, NJ 07030 USA

STEVENS

SYSTEMS ENGINEERING RESEARCH CENTER

What can INCOSE help you do?

CONNECT
Network and engage with the systems engineering community

LEARN
Enhance your knowledge through collaboration, research, and education

LEAD
Serve as an expert and thought leader to influence products and standards

PROSPER
Find career resources and improve your professional status as a systems engineer

INCOSE
International Council on Systems Engineering

www.incose.org

Visit **incose.org** to join us.

A better world through a systems approach