INSIGHT

This Issue's Feature:

graphic credit: Image generated with support from ChatGPT-4o (OpenAI) and Grok (xAI).
Visual layout and concept by Joshua Sutherland, Dean Beale, and Javier Calvo-Amodio.

SEPTEMBER 2025
VOLUME 28 / ISSUE 4

INCOSE

A PUBLICATION OF THE INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING

INSIGHT

A PUBLICATION OF THE INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING

SEPTEMBER 2025 VOLUME 28/ ISSUE 4

Inside this issue

About This Publication

INFORMATION ABOUT INCOSE

INCOSE's membership extends to over 27,000 members and CAB associates 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 six 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, digital engineering, artificial intelligence, 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.

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 six times per year. Issue and article submission deadlines are as follows:

- September 2025 issue – 1 May 2025
- October 2025 – 1 July 2025
- December 2025 – 1 September 2025
- February 2026 issue – 1 November 2025
- April 2026 issue – 2 January 2026
- June 2026 issue – 1 March 2026

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

CONTACT

Questions or comments concerning:

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

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

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

ADVERTISER INDEX September Volume 28-4

CORPORATE ADVISORY BOARD — MEMBER COMPANIES

FROM THE EDITOR-IN-CHIEF and THEME EDITORS

William Miller, insight@incose.net; **Dean Beale,** Dean.Beale@incose.net; **Joshua Sutherland,** joshua@joshuasutherland.com; and **Javier Calvo-Amodio,** Javier.Calvo-Amodio@incose.net

We are pleased to announce the September 2025 *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 theme of this issue is "Complexity and Elegance: A Call for Action." Complexity and Elegance are critical to realizing the *Systems Engineering Vision 2035*, freely accessible at https://www.incose.org/about-systems-engineering/se-vision-2035. Your editor thanks theme editors Dean Beale, Joshua Sutherland, and Javier Calvo-Amodio, and the authors for their contributions.

Complexity is an increasingly important term within systems engineering and beyond. INCOSE embeds the term complexity into the discipline's definition: "Systems engineering is an integrative approach to help teams collaborate to understand and manage systems and their complexity and deliver successful systems" (INCOSE *Systems Engineering Handbook* 2023).

Elegance and elegant solutions have been seen as the ultimate aspiration of a systems engineer when managing systems and their complexity.

Who doesn't want an "elegant solution" to their "complex problems"?

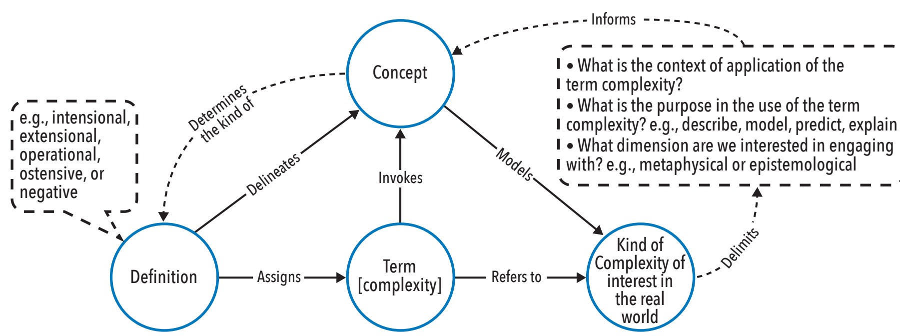


Figure 1. A category of analysis of the use of the term complexity (adapted from Rousseau, David; Julie Billingham; and Javier Calvo-Amodio. 2018. "Systemic semantics: A systems approach to building ontologies and concept maps." Systems 6 (3): 32.)

The focus of this September issue of *INSIGHT* on "Complexity and Elegance: A Call for Action" is to bring together a range of voices, who engage with these important topics and explore their use and application. So, what do we all even mean by "complex" and "elegant"?

We link the terms elegance and complexity as both being purpose and context dependent. This issue is the result of a collaboration between the Complex Systems Working Group, the Future of Systems Engineering (FuSE) program, and the Systems Science Working Group. This issue reflects four themes: 1) past uses of complexity as a term and how we got here, 2) how to evolve our use of the term complexity to address the future, and 3) current community-provided examples, resources, and perspectives on complexity.

The Complex Systems Working Group is working on developing a *Complexity Primer* version 2. Throughout that work, it has become clear that the use of the term complexity varies significantly across the systems engineering community. The Bridge Team, part of FuSE and the Systems Science working group, has connected elegance and complexity by stating that the value of systems engineering is in achieving elegant designs that resolve complex problems (*INSIGHT* 27 (2) April 2024). It is clear to us that a collaborative effort is needed to effect change in how we, as a community, use the term complexity (or its forms such as complex). We believe that the answer is not found in a single definition to rule them all, but in developing an agreed-up-upon methodology to clearly communicate the intended use of the term (e.g. describe, model, predict, explain), its context of application (e.g., domain of application, discipline, system of interest), and the kind of complexity of interest (e.g., metaphysical—what exists in the real world, or epistemological — our ability to analyze or use the system). In Figure 1 we capture the essence of our vision; the relationship between a term (what we use to communicate out concepts about complexity or elegance) and their alignment with the definitions used, that are context and purpose dependent.

This issue is long due to the richness of the topic being addressed. We have three distinct, but interrelated parts. Following is a brief description of each part.

1) The use of the term "complex" has evolved over the years, and differently in different communities. In the delivery community in which systems engineers play a significant role, and many other communities, the term "complex" is increasingly used as in uncertain, unpredictable, and unknown. This shift can be observed in the included article "Understanding Complexity: Defining a Moving Target" which tracks how the complex term usage has changed over the years.

   The challenge is that some, typically those close to delivery, recognize this shift, while others prefer to use the term as a synonym for complicated or difficult. However, what is clear is that how you approach an intricate problem that you can comprehend is very different to how you approach a problem that you cannot fully comprehend. As a result, this definitional difference is significant. The work of the Complex Systems Working Group is focused on how to address problems that you cannot fully comprehend and therefore have insufficiently certain outcomes. This is described in the "Bursting the Bubble of Complexity" article. This is important no matter what term is used and the support of the INCOSE community to progress and develop techniques that can cope with complexity or uncertainty is sought.

2) With the history and evolution about the use and meanings of the term complexity covered in part 1, we explore potential paths for evolution. We provide different perspectives (Pennotti et al., Smith, Rousseau and Billingham) on how to use or evolve the term complexity in the future. A key theme is the exploration of the relationship between the terms elegance and complexity. We present three articles that provide different perspectives about the relationship between these two terms. Each article adopts different concepts and definitions for each term. One takeaway for the readers of this *INSIGHT* issue, is that terms and their related concepts and definitions are use-dependent and context-dependent. Finally, we include a call for action to evolve systems engineering heuristics. We use heuristics to manage complexity, but if we are to embrace the new relationship between elegance and complexity, we need a robust set of heuristics we can trust. Calvo-Amodio et al. describe how the systems engineering community can engage with the Systems Science Working Group and the Heuristics Team.

3) The final section provides "practitionerlegance" in systems engineering, as defined in NASA's Theory of Elegant Systems, is not about simplicity but about achieving coherence, clarity, and robustness in the face of essential complexity. It is a disciplined approach to design that produces solutions that are effective, sustainable, and enduring. This section presents case studies from diverse domains—such as network protocols, urban transport, and astrophysics—illustrating how clarity of purpose and sound architecture can transform complex challenges into elegant systems. To support deeper engagement with complexity, it also offers a curated set of resources — books, papers, frameworks, and expert reflections — that have guided practitioners in refining their ability to recognize and design elegant solutions. Ultimately, elegance is not just found in technical solutions but also in how systems engineers frame problems, engage with complexity, and practice their craft. Seen this way, elegance becomes less a fixed outcome and more a mindset and method for navigating complexity with precision and purpose.

We look forward to hearing your feedback on this special issue.

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/publications/insight. For information about sponsoring *INSIGHT*, please contact the INCOSE marketing and communications director at marcom@incose.net. ■

INSIGHT Special Feature

Complexity Definitions Guidance in Systems Engineering

Francesco Dazzi, francesco.dazzi.13@gmail.com; **Joshua Sutherland,** joshua@joshuasutherland.com; **Gary Smith**; **Alfonso Lanza**; **Ken Cureton**; and **Dean Beale**

■ **ABSTRACT**

This article provides a brief guide addressing the inconsistent use of "complexity" and related terms across different disciplines and situations. Definition variability often leads to miscommunication, even within organizations like the International Council on Systems Engineering (INCOSE). This guide offers practical advice tailored for different audiences on how to effectively use and interpret "complexity" in transdisciplinary contexts. Rather than prescribing a single definition, it promotes a common understanding by illustrating definitional differences and providing techniques to clarify usage. This approach aims to enhance communication, and lay groundwork for a unified scientific basis for "complexity" within systems engineering.

1 WHY NAVIGATING "COMPLEXITY" DEFINITIONS MATTERS

The term "complexity" is pervasive across various domains and disciplines, frequently encompassing distinct yet related concepts such as complicatedness, emergence, difficulty, uncertainty, and chaos (hereafter referred as "related terms"). Gershenson (2008) highlighted this variety, surveying twenty-four leading figures in the field of complexity were asked "*How would you define complexity?*," a wide variety of meanings were provided. Edmonds (1999) describes the widespread overloading of the term across a wide range of fields and disciplines. This issue is evident across fields such as biology, mathematics, physics, computation, economics, software design, and management science, where each discipline may define complexity through its own lens, from self-organizing interactions to computational resources or intricate feedback loops. While individual definitions may be useful in their local contexts, their collective inconsistency can create confusion that hinders the adoption of new techniques needed to effectively manage complex systems.

In many other situations, "complexity" is used without being formally defined. For example, frameworks such as Snowden's Cynefin (2021) and the Stacey Matrix (2007) map complexity across knowledge and predictability dimensions. Many definitions remain implicit or context-bound, inferred from category descriptions rather than formal statements.

In systems engineering, this presents a unique challenge: how do systems engineering practitioners communicate clearly when the key term at the heart of their discipline means different things to different people? Indeed, this inherent variability in usage can lead to significant miscommunication and misunderstanding, even within structured organizations like the International Council on Systems Engineering (INCOSE). INCOSE, recognizing complexity as fundamental to systems engineering, embeds it in the explanation of the discipline's definition: "Systems engineering is *an integrative approach to help teams collaborate to understand and manage systems and their complexity and deliver successful systems*" (INCOSE SEH 2023). To fulfill its strategic objective, which is *advance systems engineering as the World's trusted authority* (INCOSE 2025 About), INCOSE must provide leadership on the meaning and application of "complexity."

The unmanaged variability of "complexity" definitions results in misunderstanding and vagueness, which often goes undetected. This vagueness prevents a meaningful comparison between different formulations of "complexity" across different fields of study, thereby hindering

Comparative analysis of some "Complexity" definitions and related terms

Types of Definitional Constituents	Definition:	Cambridge Dictionary	Collins Dictionary	Wikipedia – Complexity (English)	US DoD CCRP (Moffatt 2003)	MIT ESD v17 (de Weck 2011)	INCOSE Definitions (Sillitto 2019)	Complexity Primer (CSWG 2021)	Complexity Theory Proxy (Beale 2021)
		Complexity			Complex System				
Perspective	Subjective (observer-based)	✓	□	□	□	✓	□	✓	□
	Objective (observer-independent)	□	□	✓	□	□	□	□	✓
Comprehension	Unfamiliarity	□	□	□	□	□	□	✓	□
	Lack of understanding	□	□	□	□	□	✓	✓	□
Effort	Difficult	✓	□	□	□	✓	□	□	□
	Non-simple	□	□	□	□	✓	□	□	□

Figure 1. Snapshot of a table comparing different complexity definitions

progress in cross-disciplinary research and application. Navigating these definitional perspectives requires careful thought. While research often demands precision and univocal communication, diversity also offers benefits, allowing for flexibility and contextual shifts in meaning. The challenge lies in balancing the need for precision with the value of adaptability in nuanced discussions. Given these challenges, this article does not seek to develop or assert a single, universally accepted definition of "complexity." Instead, its primary objective is to provide actionable guidance on how the term, including its related terms, should be handled effectively. It aims to bridge communication gaps by:

- offering practical guidance tailored for general audiences, practitioners, and researchers
- presenting techniques for comparing definitions
- demonstrating similarities and differences across existing definitions
- proposing steps toward common terminology
- establishing a foundation for future scientific work.

This guidance is provided without advocating any meaning of "complexity," acknowledging the reality of its multiple existing interpretations.

2 TENETS ON EFFECTIVE "COMPLEXITY" TERM USAGE

This guidance is built upon four core tenets:

- avoid misuse
- acknowledge diversity
- assume misalignment
- take on communicator's responsibility.

2.1 Avoid Misuse

Before using the term "complexity," it should be ensured that its use is appropriate and necessary for communicating something distinctly different. It should be verified that it's not merely for ego or to tag work as intellectually desirable and eventually consider if an alternative term might be more effective.

2.2 Acknowledge Diversity

The wide array of definitions and meanings of "complexity" used across and within disciplines should be acknowledged and community usage recognized. Before defining the term, it should be understood how it is used within communities. The communication approach should be based on the audience.

2.3 Assume Misalignment

Given the diversity, it should be assumed that when communicating, the intended meaning of "complexity," this may not align with others' interpretations. This advice applies to both information creators and consumers, who may constructively investigate the causes of the misalignment and value the different perspectives for generating new ideas.

2.4 Communicator's Responsibility

It is the communicator's responsibility to clarify "complexity" in their work to prevent misunderstandings and foster productive discourse in a psychologically safe environment.

3 TECHNIQUES FOR CLARIFYING MEANING

A couple of exemplificative techniques are introduced to help navigate the variety of "complexity" meanings. The first one is useful in communicating and describing observed complexity, while the second one in handling and understanding "complexity" definitions.

The first one, which is among the two the most practical, is grounded on the axiom that complexity is related to the concept of system, i.e., an abstract representation of an entity as a whole showing properties emerging from the relationships and interactions between and among the system constituents. Any physical and/or conceptual entity can be modelled as a system, and complexity, as well as related terms, can be used for describing specific system aspects, such as for example the type of intricacy among system constituents. If complexity serves for characterizing a specific type of system, then any holistic techniques describing systems as a whole is suitable for describing observed complexities. A technique widely recognized in systems science and engineering is "*form-fit-function*" (Dumas et al. 1991). "*Form*" refers to the structural characteristics of a system, i.e., "what it is." "*Fit*" refers to how well a system relates to its context ensuring that all interfaces work together harmoniously. "*Function*" refers to the system's behavior, describing the actions or purpose that the system should perform, i.e. "what it does." It is worth noting that the "form-fit-function" (FFF) technique is remarkably similar to "system architecture" from systems engineering. "Form" is equivalent to "physical architecture", "fit" to "system integration and interfaces", and "function" to "functional architecture." However, the "form-fit-function" technique is suggested for describing observed complexity because a proper use of the system architecture could depend on the usage of dedicated tools, which might limit communication efficiency between people from different domains. "*Form–fit-function*" makes complexity conversations repeatable and translatable across disciplines, without forcing premature convergence on a single definition.

The second technique is more theoretical than the first one. It is based on breaking down "complexity" definitions into definitional constituents. (Beale et al. 2024) builds a table to compare six alternative natural language definitions of complexity, by noting if any particular element of complexity is included or not included in the definition. See a snapshot of a similar table in Figure 1.

The approach of (Beale et al. 2024) make the differences of the definitions significantly less obscure than having the reader parse each definition individually. This technique is useful for performing comparative analyses that break definitions down into constituents such as structure, behavior, or comprehension. This helps revealing:

- which "complexity" aspects are considered, and which are explicitly excluded
- the importance given to different "complexity" aspects
- which "complexity" aspects are completely ignored.

This technique also supports the development of "complexity" definitions that fit coherently into their corresponding business glossaries, also known as technical vocabularies. Even more, the application of this technique to "complexity" related terms might contribute to the endeavor of establishing an "ontology of systemology," as proposed in Rousseau et al. (2018).

4 AUDIENCE-DRIVEN COMMUNICATION APPROACHES

An effective communication relies on the ability to adapt the communication approach based on the type of audience and specific situation. Specific guidance is then tailored for three primary audiences, i.e.:

- people who use it in general / casual situations
- people who use it in professional situations within practitioner communities
- people who use it in research.

4.1 General / Casual Usage

In everyday conversation, "complexity" and related terms are often used interchangeably. When communicating to a general audience, information creators should focus on clarity and accessibility. They should adapt definitions to emphasize relevant and easy-to-understand aspects, avoiding excessive detail. Examples or stories should be considered for communicating the intended meaning.

Information consumers should avoid being overly literal. Instead of assuming a specific technical meaning, they should focus on grasping the broader concept being conveyed. This flexible approach facilitates clearer communication in casual contexts.

Simple metaphors or analogies grounded in "*form–fit-function*" can support clarity, e.g., a traffic system (form) that fails during a snowstorm (fit), disrupting city-wide mobility (function).

4.2 Practitioner Community Usage

The practitioner community faces the greatest risk of miscommunication due to their likely familiarity with a specific, context-driven definition that may differ significantly from those in other disciplines. Information creators should clearly state the definition upfront and maintain consistency. They should illustrate definitions with practical examples to contextualize the term within real-world applications. They should explicitly clarify any critical concepts like uncertainty or emergence, ensuring that the content aligns with the stated definition to prevent misunderstandings.

Information consumers should be aware of the specific context in which "complexity" is used, as definitions vary across projects or organizations. They should request examples to understand how "complexity" manifests in that particular context. If a definition or suitable example is lacking, they should ask for clarification to bridge understanding and minimize miscommunication risks.

An additional consideration could involve being explicit about which aspect of complexity is being emphasized:

- form (system internal structure),
- fit (system context and external environment), or
- function (system behavior).

It is especially useful for practitioners to indicate whether complexity stems from form limitations (e.g., legacy architectures), fit dynamics (e.g., regulatory shifts), or emergent function (e.g., unexpected user behavior). Highlighting this focus provides a helpful point of reference when communicating across disciplinary boundaries. For instance, a software engineer might emphasize "*form–function*" complexity, while a policymaker may be concerned with "*fit-related uncertainties.*"

4.3 Research Community Usage

The research community demands a more formalized and precise approach, often using mathematical, technical, or empirical methods to illustrate key aspects. Information creators should adopt technical definitions complemented by formal, often quantitative examples (e.g., structural complexity quantified with matrix formulas). They should provide clear citations to allow readers to trace the origin and application of a specific definition, ensuring consistent application of definitions throughout the work.

Information consumers should prioritize identifying and understanding the specific technical definition employed in each source, paying attention to its context and implications for the study's findings. They should be prepared to inquire about unclear definitions or methods, especially if their alignment with the study's analysis is not transparent.

Researchers should clarify the definition requirements. This implies that they should structure the intended meaning of "complexity" using a range of definitional constituents or suitable mathematical constructs. They might also consider mapping these definitional constituents explicitly to the dimensions of "*form-fit–function*." For instance, structural interactions suggest complexity in form, unpredictable outcomes may indicate function-related emergence, while evolving external demands often signal fit-related instability. Researchers can benefit from structuring their definitions around "*form-fit-function*," either implicitly or explicitly, as a way of making underlying assumptions traceable and more comparable across domains.

Subsequently, researchers should communicate and standardize the definition. Once the definition is clarified, researchers should explain the rationale behind the chosen definition, recognizing that others may have different, unaligned interpretations. This fosters feedback and enables iterative refinement. Crucially, researchers must consistently adhere to their defined meaning throughout their work.

5 BENEFITS OF A CLEAR COMPLEXITY LEXICON

The adoption of a clear and consistent lexicon for "complexity," guided by the tenets, techniques, and approaches outlined respectively in sections 2, 3, and 4, yields significant benefits across systems engineering practice and research.

5.1 Enhanced Communication

Systems engineering practitioners can more effectively explain project challenges, limitations, and strategic choices to diverse stakeholders, including management, clients, and regulators. This transparency helps manage expectations, builds trust, and facilitates better resource allocation and risk acceptance. An agreed understanding of terms like "complicated" versus "complex" minimizes ambiguity and reduces misunderstandings among stakeholders. By referencing "*form-fit-function*" explicitly, systems engineering practitioners can communicate system challenges in a way that is intuitive: for example, showing how a legacy form may not adapt well to new regulatory fit, or why a function is not emerging as expected.

5.2 Accelerated Organizational Learning

A consistent lexicon provides a standardized framework for analyzing project successes and failures related to complex challenges. This common language allows organizations to systematically capture and build a cumulative body of knowledge on managing different types of complex endeavors, leading to continuous improvement in systems engineering practice. This aligns directly with INCOSE's role in *developing and disseminating the transdisciplinary principles and practices that enable the realization of successful systems* (https://www.incose.org/about-incose) and providing «*impactful guides for the community* (https://www.incose.org/publications).

5.3 Advancing the State of Practice

By providing a shared, practical guide for understanding and discussing complexity, this work directly contributes to INCOSE's overarching objective of *advancing the state of practice in systems engineering and to close the gap with the state of the art* (INCOSE n.d. About *INSIGHT* Magazine). It transforms abstract theoretical understanding into actionable wisdom for practitioners, enabling them to apply cutting-edge concepts in their daily work. The adoption of the techniques described in section 3 strengthens this transformation by providing a universally applicable scaffold that clarifies where complexity lives and how it evolves.

The acceptance of this guide by INCOSE signifies INCOSE's endorsement of a standardized, practical approach to managing complexity, influencing training, education, and future industry standards.

6 CONCLUSIONS

The modern systems engineering landscape is profoundly shaped by complexity. Understanding and consistently applying accurate definitions of "complexity" and related terms, and, crucially, understanding the variability of these terms, is an indispensable skill for every systems engineering practitioner.

This article provides a practical guide for identifying, characterizing, and effectively communicating about complexity throughout the system lifecycle. By embracing this guide, systems engineering practitioners can move beyond intuitive, often inconsistent, interpretations of complexity to a more structured, analytical approach. This leads to more predictable outcomes, reduced project risks, and ultimately, the successful realization of resilient and adaptable systems. As an additional foundation, the "*form-fit–function*" triad offers a unifying structure grounded in system science. It enables systems engineering practitioners to diagnose the roots of complexity, whether in internal form, contextual fit, or emergent function, and to apply appropriate approaches. When treated holarchically, "*form-fit–function*" scales across levels of abstraction and hierarchy, supporting the navigation of complexity in both technical systems and socio-technical environments.

Systems engineering practitioners are strongly encouraged to integrate this lexicon and framework into their daily practice, team discussions, and project documentation. Sharing this guide within organizations can foster a common understanding across diverse teams and stakeholders, thereby enhancing overall project communication and collaboration. This guide represents a foundational step in empowering systems engineering practitioners to navigate the challenges of an increasingly complex world. As systems continue to grow in scale, interconnectedness, and dynamic behavior, a shared, nuanced understanding of complexity will become even more critical.

This clarity equips systems engineering practitioners to lead effectively, innovate responsibly, and contribute meaningfully to INCOSE's mission *fosters systems engineering knowledge exchange, application, education, and research* (https://www.incose.org/about-incose), ensuring that the discipline remains at the forefront of addressing the world's most daunting challenges. ■

REFERENCES

- Beale, D., F. Dazzi, and T. Tryfonas. 2024. "Unweaving the Definitions of Complexity." *IEEE Transactions on Systems, Man, and Cybernetics* 54 (2): 682-692.
- Dumas, Angela, and Henry Mintzberg. 1991. "Managing the Form, Function, and Fit of DESIGN." *Design Management Journal* 2 (3): 26-31. Summer. https://doi.org/10.1111/j.1948-7169.1991.tb00573.x.
- Edmonds, Bruce. 1999. "Syntactic Measures of Complexity." PhD diss., Manchester Metropolitan University (Manchester, GB-ENG).
- Gershenson, Carlos. 2008. *Complexity: 5 Questions*. Automatic Press.
- INCOSE. n.d. "About *INSIGHT* Magazine." https://incose.org/insight.
- INCOSE. 2025. "About INCOSE." https://www.incose.org/about-incose. Accessed 17 June.
- INCOSE SEH. 2023. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities* Version 5. Revised and edited by D. D. Walden, T. M. Shortell, G. J. Roedler, B. A. Delicado, O. Mornas, Yip Y. S., and D. Endler. San Diego, US-CA: International Council on Systems Engineering. Published by John Wiley & Sons, Inc.
- Rousseau D., J. Billingham, and J. Calvo-Amodio. 2018. "Systemic Semantics: A Systems Approach to Building Ontologies and Concept Maps." *Systems* 6 (3):32. https://doi.org/10.3390/systems6030032.

- Snowden, D. (author), R. Greenberg (editor), and B. Bertsch (editor). 2021. *Cynefin –Weaving Sensemaking into the Fabric of our World*, Version 2 of Edition 1. The Cynefin Company: Cognitive Edge. ISBN: 978-1735379906.
- Stacey, R. D. 2007. *Strategic Management and Organisational Dynamics: The Challenge of Complexity to Ways of Thinking About Organisations*, 5th Edition: Pearson Education Ltd. ISBN-13: 978-0273708117.

ABOUT THE AUTHORS

Francesco Dazzi is a senior technologist at the Italian National Institute for Astrophysics (INAF), president 2022-2023 of AISE (INCOSE Chapter Italy), and member of the INCOSE Institute for Technical Leadership. He has more than 20 years of experience mainly dedicated to the construction of ground-based telescopes and technical management of scientific projects. He has a degree in physics, a PhD in mathematics and physics, and more than 250 publications including journal articles and conference proceedings. In 2018 he obtained the CSEP (certified systems engineering professional) certification.

Joshua Sutherland is co-chair of INCOSE's Complex Systems Working Group (CSWG) and co-lead of the Future of Systems Engineering (FuSE) scientific foundations initiative. Joshua is the founder and president of Sutherland Systems Engineering Enterprise (SSEE) Ltd, a consultancy and media group specialising in complex systems and technology development. His team delivers projects for global clients including Ford Motor Company, the European Space Agency, and leading technology firms in Silicon Valley and Japan. Joshua is also a fellow at the University of St. Gallen's Center for Aviation and Space Competence and hosts the world's leading podcast in the systems engineering space.

Gary Smith is a senior expert systems engineer at Airbus Defence and Space, where he leads the architecture of engineering processes and the digital transformation of the division. He is an INCOSE certified expert systems engineering professional (ESEP) and a senior editor of the Systems Engineering Body of Knowledge (SEBoK). A long-standing advocate for integrating system science with engineering practice, Gary has served as vice president for systems practice at the International Society for the Systems Sciences (ISSS) and as the society's liaison to INCOSE. He has contributed extensively to the development of coherent frameworks for systemic inquiry, practice, and education, bridging complex domains from aerospace and defence to healthcare and sustainability. Now serving as past president of the ISSS, Gary continues to champion collaboration across disciplines and professional societies. He is a leading contributor to the emergence of a unified system science enterprise, focused on establishing shared foundations that support both scientific understanding and practical application. His work emphasizes the triad of fit, form, and function as a basis for achieving coherence, managing complexity, and cultivating systemic elegance.

Alfonso Lanza holds engineering degrees from Imperial College London and the University of London.

Kenneth L. Cureton holds a BS in physics from California State University Los Angeles and an MS in systems architecting and engineering from USC, where he has served as a part-time lecturer since 1996. He retired from Boeing (Rockwell) after 29 years, with additional experience spanning 16 years in commercial and government sectors. His expertise includes enterprise systems architecting, system-of-systems modeling and integration, complex networked systems interoperability, resilience engineering, and avionics design for manned and unmanned space systems. Cureton chairs the INCOSE Resilient Systems Working Group and serves on AIAA Space Settlement and Systems Engineering Technical Committees. He previously co-chaired the IEEE SMC MBSE Working Group and served as technical council chair emeritus for the NCO Industry Consortium.

Dean Beale has a PhD in organisational complexity, a Master's degree in solid state physics, and is a chartered engineer with the IET. He spent his early career leading R&D at DRA, DERA, and QinetiQ, and was a visiting lecturer at the University of Birmingham. This work naturally led to systems engineering roles and research in organisational complexity at the University of Bristol. He is chair of the International Council On Systems Engineering (INCOSE) Complex Systems Working Group and serves as INCOSE deputy assistant director for international symposium technical operations coordination.

Elegance and Complexity

Michael D. Watson, michael.watson@incose.net; and **Andrew T. Fried,** fried@alumni.stanford.edu

Elegance is not often thought about when sitting in engineering discussions about systems concepts, requirements, analysis, design, or verification. Colloquially, elegance describes aesthetics in appearance or execution, often in social settings. It entails characteristics such as intricacy, beauty, and a sense of well-planned, flawless execution. Considering these characteristics, engineering of systems is immediately seen. Systems engineering deals with intricate interactions of the system, internally and externally, and intricate relationships within organizations, i.e., social settings, and social interaction in use or operations. The idea of beauty is more ethereal for systems, yet a well-designed and assembled system has a beauty to it. This may be as much in the internal system design as in the external aesthetics. Consider the DC-3 cargo plane, P-51D Mustang fighter aircraft, Saturn V rocket, F-1 race cars, Micra™ Transcatheter Pacing System (TPS) (a trademark of Medtronic), and numerous other examples from aerospace, automotive, ships, and medical devices. These examples all share the characteristics of elegance in they are well designed appropriately incorporating and managing their intricate relationships, have responsive and efficient user interfaces, and have a beauty about their appearance and functionality. Clearly, elegance is an attribute of a well-engineered system.

Robert Frosch first introduced the idea of elegance in systems engineering. He stated that the proper goal of systems engineering is to produce an elegant design. He noted that he often got no response when he asked systems analysts, "Is it an elegant solution to a real problem?" They did not understand the question. Michael Griffin emphasized this in his paper "How Do We Fix System Engineering" (Griffin 2010).

This idea of system elegance was elaborated in the NASA Technical Publication, "Engineering Elegant Systems: The Theory of Systems Engineering" (NASA 2020a, 2020b). Elegance is something you know when you see it, but is not something easily defined, particularly in the sense of a system. Webster defines elegance as a "dignified richness and grace." This articulates an attitude of intent and a social response to the system. This definition identifies key system attributes. 'Dignified grace' conveys a notable ease of use or operation in a variety of applications. 'Dignified richness' conveys a notable robustness in application, a full achievement of the system intent, and a satisfaction of intent not fully specified. A term that provides further help with this definition is concinnity. Webster defines concinnity as 'a skillful arrangement of parts, harmony, and elegance.' This conveys the idea of a well-organized system with skillfully defined system interrelationships. System aesthetics are accounted for in the idea of richness, grace, and harmony. An efficiency in the system layout and construction is also seen in the 'skillful arrangement of parts, harmony' of the system. A well-structured system is an efficient system. Perhaps one can state a definition of system elegance as 'a system that is robust in application, fully meeting specified and adumbrated intent, is well structured, and is graceful in operation' (NASA 2020a, 2020b).

Complexity brings in a whole new need for elegance. Complexity is not realized without elegance in the system design and implementation. Complexity cannot be forced into existence. The characteristics are too intricate to be realized in a haphazard manner. Simple and complicated systems can be decomposed and deal with uncertainty through compartmentalization of relationships seen in interface definition. As systems move from complicated to complex, the ability to approach systems in this manner becomes ineffective. If you approach complexity from a compartmentalized understanding, the characteristics and system responses can appear to be arbitrary or random. This leads to challenges in implementing complex systems, requiring an appreciation of the elegance required for the design in order to achieve the realization of the system.

Complex systems have 14 characteristics defining their complexity. These characteristics were defined by INCOSE at the International Symposium in 2019 (Watson et al. 2019) and later included in the A Complexity Primer for Systems Engineers Revision 1 in 2021 (INCOSE 2021). These characteristics are diversity, connectivity, interactivity, adaptability, multiscale, multi-perspective, behavior, dynamics, representation, evolution, unexpected emergence, disproportionate effects, indeterminate boundaries, and contextual influences. Note that general system emergence is a characteristic of all systems. Unexpected emergence is a characteristic more prevalent in complex systems. Complexity in systems is a gradient along these 14 characteristics. Systems can exhibit complexity in a few characteristics but may show simple or complicated characteristics in others (NASA 2020a, 2020b).

Considering these characteristics of complexity and the gradient that can exist in each of these characteristics, system definition and design becomes intractable following methods based only on system decomposition. As system decomposition vanishes with increased complexity in a characteristic or set of characteristics, the need for a more refined, intricate approach becomes necessary in systems engineering of the system. This need brings in elegance in complex system design. Engineering the system requires a detailed understanding of the system physics. Logical constructs, operations, and the engineering organizations social interactions in order to achieve an effective complex system design. This comprehensive understanding is necessary for elegance in the system design. The examples of elegant systems listed above have limited or no complexity and most were designed by decomposition methods. Their elegance arrived at through the experience of the designers, seeking detailed feedback of the operators (e.g., pilots) and/or extensive test campaigns. Systems with high complexity gradients across their characteristics do not lend themselves to these practice approaches. The elegance must be considered early in the design phase to ensure a system can be constructed or grown or evolved to achieve the end goals in behavior and use.

Systems innovation is required to achieve elegance in a complex system and modeling of such a system is paramount to systems innovation, because it equips the systems engineer with a better understanding of the system. As systems engineers, we must constantly be asking about causality and if there is another way to design the system that would make it more elegant. Furthermore, each system decision during the development must be considered in light of the principles of elegance in a complex system to avoid creating unnecessary or unintended consequences in the system. For all of these reasons, elegance in systems engineering is required to realize systems with higher levels of complexity.

Systems today are driven by high levels of software functionality, autonomy, and system interdependencies (i.e., systems of systems). Artificial intelligence, providing more independent behavior and system action, is also emerging as a capability of future systems. With these confounding factors of complexity, i.e., multipliers of complexity (NASA 2020a, 2020b), elegance in system design is becoming a necessity to realize the system, not something to be achieved arbitrarily. As we move towards higher gradients of complexity, elegance in our approach to systems engineering is a necessity! ■

REFERENCES

- Frosch, R.A.: "A Classic Look at Systems Engineering," in Readings in Systems Engineering, F. T. Hoban and W. M. Lawbaugh (eds.), NASA SP–6102, Washington, DC, 1993.
- Griffin, M. D., "How Do We Fix System Engineering?" 61st International Astronautical Congress, Prague, Czech Republic, 27 September – 1 October 2010.
- INCOSE. 2021. "A Complexity Primer for Systems Engineers", Revision 1. Edited by M. D. Watson, D. Beale, and D. McKinney. November.
- NASA. 2020a. "Engineering Elegant Systems: Theory of Systems Engineering." NASA/TP–20205003644, edited by M. D. Watson, B. Mesmer, and P. Farrington, NASA Marshall Space Flight Center, Huntsville, US-AL, June.
- NASA. 2020b. "Engineering Elegant Systems: The Practice of Systems Engineering." NASA/TP–20205003646, edited by M. D. Watson, B. Mesmer, and P. Farrington. NASA Marshall Space Flight Center, Huntsville, US-AL, June.
- Watson, M., D. McKinney, R. Anway, L. A. Rosser, and J. MacCarthy. 2019. "Appreciative Methods Applied to the Assessment of Complex Systems." INCOSE International Symposium, Orlando, US-FL.

ABOUT THE AUTHORS

Michael D. Watson is the International Council on Systems Engineering (INCOSE) president-elect and the senior systems engineering director for the Leidos Defense Systems Sector. He is responsible for the application and execution of systems engineering across the defense systems sector. He is focused on the advancement of systems engineering and the realization of more complex systems through the application of systems engineering. He previously served as INCOSE chair of the Complex Systems Working Group. Dr. Watson retired from the National Aeronautics and Space Administration (NASA) Marshall Space Flight Center (MSFC). He worked on SpaceLab, ISS, Ares I, and SLS. He led the NASA systems engineering research consortium defining advanced systems engineering approaches and basis. Dr. Watson graduated with a BSEE from the University of Kentucky in 1987. He obtained his MSE in electrical and computer engineering (1996) and his PhD in electrical and computer engineering (2005) from the University of Alabama in Huntsville.

Andrew Fried is a lead systems engineer and technical fellow at Medtronic in Pelvic Health /Neuromodulation, with two decades of experience working both in biomedical engineering and in aerospace. He leads MBSE-driven architectures for implantable neuromodulation systems, wireless recharging platforms, and patient programmers, emphasizing elegance, cybersecurity, and human-centered design. Andrew holds 22 U.S. patents across 16 families. A proponent of inclusive innovation, he mentors cross-functional teams and collaborates across disciplines. He has served as an adjunct professor of systems engineering at the University of St. Thomas and the University of Minnesota, and founded Meaningful Systems LLC to advance and share systems thinking and innovation practices. Andrew earned an MS in electrical engineering from Stanford University and is an active member of the INCOSE North Star Chapter.

Bursting the Bubble of Complexity:

Reflections on the activities of the INCOSE Complex Systems Working Group (CSWG)

Dean Beale, dean.beale@Incose.net; **Rudolph Oosthuizen,** rudolph.oosthuizen@up.ac.za; **Dorothy McKinney,** dorothy.mckinney@incose.net; **Francesco Dazzi,** francesco.dazzi@incose.net; and **Joshua Sutherland,** joshua@joshuasutherland.com

■ **ABSTRACT**
We have entered an era of rapid change and increasing uncertainty. The biggest mistake we can make as systems engineers is ignoring this change. The term "uncertainty" is deeply connected to complexity for many communities, including INCOSE's Complex Systems Working Group (CSWG). As uncertainty or complexity increases, our experience, and indeed logic, suggests that the practices and techniques developed for a world of sufficient certainty are no longer enough, no matter how gifted the engineer. At this point, it is essential to change our mindset from "I know enough" to "I know I am wrong about something". This mindset shift triggers a desire and need for continuous learning, new practices and techniques which embrace different viewpoints, learning through failure, and results in flexible and adaptable systems. The work of the CSWG, described in the article, is to identify and create suitable practices for the practitioner systems engineer in this new, uncertain, and complex age. We are aiming to "burst the bubble of complexity" and enable engineers to deal effectively with uncertainty and complexity. But the pace of change is fast, the complexity landscape is vast, and the tradecraft still emerging. Hence, the only way to address this complexity challenge sufficiently is to recruit communities of experts with diverse views who can work collaboratively towards these common aims.

INTRODUCTION

In this article, we share our view of key changes in progress in the world, what complexity has to do with these changes, lay out a framework for identifying when our systems engineering approach needs to change, and describe the specific products and activities the Complex Systems Working Group (CSWG) has in progress to help systems engineers deal more effectively with these changes.

It is common to claim that systems are complex, and many professionals (engineers and others) gain recognition from publicizing their work with complex systems. The INCOSE CSWG feels that it is important to "burst the bubble" of complexity claims, and note some key realities: (1) many of the systems described as complex are just very complicated — and often the word "complex" is used as a synonym for "difficult" and (2) even systems which are complex are not uniformly complex— they have areas of complexity, and usually also areas of intricacy (complicatedness) and often areas of simplicity. What we mean by complexity is described below.

THE AGE OF UNCERTAINTY?

We live in a world undergoing rapid and unprecedented change. The once predictable global order is now characterized by volatility, disruption, and complexity. As Justin Trudeau, former Prime Minister of Canada, famously stated: "The pace of change has never been this fast, yet it will never be this slow again." This acceleration is both measurable and impactful.

Evidence of this transformation is widespread. The International Monetary Fund's Uncertainty Index, which tracks economic policy uncertainty, has shown a significant rise since the 2008 financial crisis, indicating that heightened uncertainty has become the new normal. This

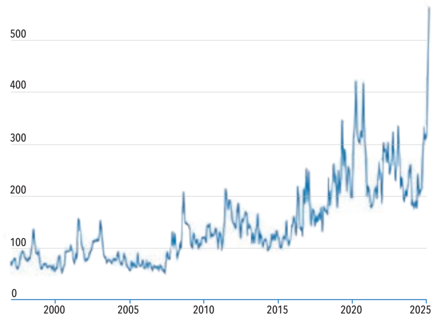


Figure 1. Monthly global economy policy uncertainty index (Baker et al., n.d.)

has recently peaked at more than 6 times more uncertainty than the 2007 level before which many delivery techniques were first developed.

Morris (2018) in *The Big Shift* analysed global trends from the Industrial Age to the present, listing 83 exponential trends that could change everything — from artificial intelligence (AI) and automation to climate disruption and geopolitical fragmentation. These trends challenge legacy thinking and indicate the Big Shift is one from certainty to uncertainty, and which demand adaptive strategies.

INCOSE *Systems Engineering Vision 2035* (2021a) identifies a future shaped by accelerating technological disruption, global interconnectedness, and rising systemic complexity. Rapid advances in AI, autonomy, and data-rich environments are transforming engineered systems, pushing beyond traditional linear models into dynamic, adaptive, and unpredictable domains. Societal and environmental pressures such as climate change, demographic shifts, and ethical concerns demand more sustainable, resilient, and inclusive solutions. *Systems Engineering Vision 2035* calls for a significant change in systems engineering practice, emphasizing new mindsets, interdisciplinary collaboration, and continuous learning to address challenges in an increasingly uncertain world. Systems engineers must evolve their methods, tools, and thinking to thrive in this complex environment.

This transformation recognizes a pivot toward a world dominated by complexity, interconnectivity, and systemic fragility. Political polarization, technological disruption, legal ambiguity, and environmental degradation now define the operating environment for governments, industries, and communities alike. Systems engineers must acknowledge that the world has changed and continue to adapt accordingly.

Increases in uncertainty are manageable when there remains sufficient causality between cause and effect. However, when uncertainty is high, this causality diminishes, changing everything. To cope with this level of uncertainty, we need to change our mindset, techniques, and skills to survive and thrive in the new post-industrial age. A simple application of PESTLE (political, economic, sociological, technological, legal, environmental) analysis to the global order reveals cascading systemic risks and unpredictable feedback loops in what Bennett and Lemoine (2014) characterize as a rising volatile, uncertain, complex, and ambiguous (VUCA) world, traditional planning, design, and leadership approaches are no longer sufficient in this age of uncertainty.

Consequently, the biggest mistake we can make as a systems engineer's community is to ignore this change. We need to recognize that the challenges we once faced are unlikely to be the same challenges we will face in the future. Acknowledging the shift to post-industrial uncertainty means recognizing that we must also change.

Table 1. PESTLE analysis of the global world order (OpenAI 2025)

Category	Key Trends and Issues
Political	Geopolitical fragmentation (e.g., U.S.–China, Russia–West) Weakening of multilateral institutions Rise of authoritarianism and populism
Economic	Persistent post-COVID inflation and sovereign debt Economic nationalism and global supply chain reconfiguration
Sociological	Aging in developed nations vs. youth bulges in developing ones Increased migration and political polarization Inequality and mental health crises
Technological	AI transforming jobs, security, and governance Rise of autonomous systems and cyberwarfare threats
Legal	Erosion of international legal norms Growing demand for regulation of AI, data privacy, and tech monopolies
Environmental	Intensifying climate events and resource insecurity Energy transition disrupting economic and geopolitical norms

THE LINK BETWEEN UNCERTAINTY AND COMPLEXITY

This uncertainty shift is directly linked to complexity as defined by the INCOSE CSWG in "A Complexity Primer for Systems Engineers" (INCOSE 2021b):

"A complex system has elements, the relationship between the states of which are weaved together so that they are not fully comprehended, leading to insufficient certainty between cause and effect."

In contrast:

"A complicated system has elements, the relationship between the states of which can be unfolded and comprehended, leading to sufficient certainty between cause and effect."

These definitions reflect that the cause of this shift away from certainty (associated with simple and complicated systems) to insufficient certainty (uncertainty or complexity) is largely due to hyper-connectivity, or relationships, and our ability to understand the impact of all the connections.

The difference between how we manage sufficiently certain systems and systems that are insufficiently certain (uncertain) is significant. In the sufficiently certain system, a leader can be an integrator of knowledge and pass down helpful directions, confident that he/she will be sufficiently right and expecting direction to be followed and delivered with no adverse consequences. However, if the system is uncertain, then that exact same behaviour is the wrong thing to do. Often this need to shift to handling complex, or insufficiently certain, tasks go unnoticed.

Consequently, there appears to be a hidden cliff edge where a completely new mindset and set of tools are required. This hidden cliff edge has been referred to by Beale, Dazzi, and Tryfonas (2023) as the Threshold of Complexity, see Fig. 2.

The Threshold of Complexity can be understood using familiar terms by conducting a simple thought experiment. A system with a low level of intricacy often can be considered simple. As it becomes more intricate and the number of elements, and more importantly, the connectivity between the elements increases, then there comes a point where the problem might be compli-

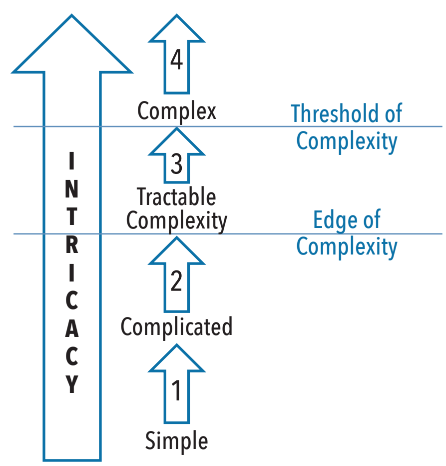


Figure 2. Adapted unified definition of complexity model, Beale, Dazzi, and Tryfonas (2023), that suggests how intricacy can connect key terms around the definition of complexity (Crown Copyright ©2023)

Table 2. Table to indicate how to shift mindset to cope with additional complex challenges

With a mindset alert to complex challenges we: Do more of	Do less of
Characterize and localize complexity	Treating all issues as either entirely complex or entirely complicated
Cohere and communicate compelling community vision and associated strategy	Concentrate on short-term deliverables that motivate only a small part of the community.
Acknowledge fallibility and see errors as learning opportunities	Striving for perfection and ignoring small errors
Practice humility and stay open to new perspectives	Relying just on personal expertise and past successes
Select approaches based on current context	Applying proven tools or frameworks without assessing fit
Develop psychological safety	Assuming others will speak up if they disagree
Seek diverse perspectives	Relying only on the most experienced or technically proficient voices
Drive feedback, experiment, and learn continuously during development	Upfront design and planning which depends on minimal change mid-project in areas of complexity
Proactively establish feedback loops	Expecting feedback to emerge naturally through normal work cycles
Decide and act early, to learn through iteration, and avoid analysis paralysis	Delay initiation of tasks by seeking certainty in data and plans.
Iterate with small experiments	Counting on big, bold changes to make faster impact, without allowing for iteration afterwards
Observe and orientate quickly before acting	Jumping straight into solution mode without fully assessing the landscape
Design systems that grow and adapt post-delivery	Relying on requirements mgmt. to prove that the right system was built right
Developing adaptable and resilient systems	Optimize only for functionality

cated or difficult but sufficiently comprehended with effort. As intricacy increases further, there must come a point where standard approaches and methods will fail to aid comprehension of the whole situation. At this point, systems engineers need to do something different or innovative. This point has been named the Edge of Complexity, and leads to entry into the tractable complexity zone, where many systems engineers operate. However, as intricacy increases further, there must be a point where all the tools, techniques, and innovations available fail to aid comprehension and make the system sufficiently certain, i.e., the uncertainty is greater than the acceptable risk appetite. This is the Threshold of Complexity; above this threshold the system cannot be sufficiently comprehended and any solution developed is highly likely to have unexpected emergence. This leads to a key question: how is it possible to engineer a system that cannot be fully comprehended?

With a complex system, it is important to accept the uncertainty and recognise it is impossible to be sufficiently right. Recognising fallibility creates the need to seek diverse views in decision-making and to build living systems and/or systems with proactive feedback so that errors can be spotted and resolved rapidly with minimal impact. The approach to solving complex or uncertain problems is more expensive, so it should only be applied when needed, but it is not as expensive as catastrophic failure (now common with large projects) when the complexity is ignored. Table 2 summarises the difference between a traditional systems engineering mindset and one better suited to dealing with complexity. (Please note that this mindset change does not mean that standard systems engineering tools, approaches, and practices for the complicated portions of a complex system are not valuable. But it does mean recognising that there is a different toolset required for the complex portions of the problem.)

COMPLEX SYSTEMS WORKING GROUP (CSWG)

This article aims to relate this change to systems engineers to ensure that we and our tradecraft evolve to meet the demands of directing the global organizations we are part of, enabling them to thrive in this new post-industrial VUCA world.

The CSWG is a community within INCOSE, established in response to the increasing challenges presented by the new age of uncertainty. The CSWG unites a diverse body of systems engineering practitioners and researchers from around the globe, all working to enhance the engineering community's ability to understand and respond to complexity across technical and organizational contexts.

The group emphasizes that a fundamental shift in mindset is required — systems engineers must adapt to a new paradigm, embracing more adaptive, exploratory, and collaborative approaches in the face of uncertainty and change. This includes recognizing when they are operating near the "Threshold of Complexity," where conventional, linear methods may no longer be effective. In light of these challenges, the CSWG has identified several key changes the community must undertake:

- **Recognize and Respond to the New Paradigm:** Engineers must remain alert to emerging complexities and actively shift their mindset to address them.
- **Establish New Scientific Foundations:** There is a need to build a trade-

craft addressing complexity, beginning with the articulation of clear terms and heuristics. These heuristics will form the early steps toward establishing enduring principles, creating a scientific derived foundation for handling complexity or uncertainty in systems engineering.

- **Expand the Toolset:** Many of the tools that have served us well in the past are likely to be inadequate — or even counterproductive — if they are the only ones applied in the future. New tools must be developed and adopted to help recognize the Threshold of Complexity and navigate around it.
- **Foster Community-Wide Psychological Safety:** Progress in complex systems engineering requires a culture that accepts the inevitability of being wrong at times. Creating a safe space for constructive dialogue and mutual support is essential for the community to learn and grow.

The CSWG's efforts aim to help systems engineers more effectively recognize, characterize, and manage complex systems where traditional, linear engineering methods may no longer be sufficient. The CSWG's international and interdisciplinary membership provides a wealth of both practical experience and academic insight, ensuring its outputs remain innovative and grounded. The group has spearheaded initiatives such as The Complexity Primer for Systems Engineers (INCOSE 2021b), curated essential heuristics (Beale et al. 2022, 2023) progressing them with the Bridge Team, and actively contributed to INCOSE's Future of Systems Engineering (FuSE) initiatives—all aimed at advancing the profession to meet the demands of our increasingly complex uncertain world. Below is a list of activities and products being actively worked on by the CSWG to achieve its aims.

1. **The Complexity Primer for Systems Engineers v2**

One of INCOSE's most potent instruments for sharing insights across its community is the primer series of documents. The Complexity Primer for Systems Engineers Revision 1 (2021b) provides valuable insight into understanding and addressing complex problems. However, one of the by-products of the hyper-connected information age is that there is too much information to be consumed, meaning the appetite for reading 30 pages of dense academic text is reduced. So, a transformational approach was required following INCOSE's encouragement to create shorter, catchier primers and seeing the wealth of new insight that could not be included in such a document. Rather than building

Figure 3. Complexity Primer for Systems Engineering v2 (in development)

a primer that was rich academically but rarely read, we seek to create a primer that is much more consumable by systems engineering practitioners and stakeholders to reach the broadest possible community and change their mindset.

The primer needs to grab the casual reader's attention and start readers on the journey of mindset change. This was a new direction for the CSWG, and we recognized and encouraged others to engage with the task focus on changing mindsets rather than providing rich academic insight, while providing quick access to more detailed information using QR codes to enable the consumer of the primer to continue the journey of discovery.

2. **Exploring Scientific Foundations**

The CSWG Scientific Foundations team has made key progress in understanding the history and evolution of the term complexity, clarifying its varied usage, and offering practical guidance on its usage across audiences. Drawing from INCOSE's expansive Heuristics database, several papers have consolidated a core set of Heuristics, with ongoing efforts to elevate at least seven of them into candidate Principles for managing complexity in engineered systems. Recognizing that complexity remains a contested and overloaded term, we are comparing interpretations across organizations, such as MIT, NASA, and the CSWG and developing shared ontologies (Sutherland 2025). This work supports INCOSE's *Systems Engineering Vision 2035* ambition to move from application-derived principles to those with scientific and mathematical grounding. By formalizing terms, experimenting with hypotheses like the "Conservation of Complexity," and linking heuristics to emerging theory, we aim to lay a foundation that enables a scientifically grounded systems approach in an increasingly complex world."

3. **Creating New tools**

To aid engineers in understanding and navigating the threshold of complexity, the Pleko framework has been developed, (Beale et al. 2025). It allows users to consider the type of problem being addressed by considering the intricacy of the system and the confounding factors (Watson et al. 2019). It then provides checklists of approaches to remove unnecessary complexity to ensure correct categorisation. Combined with the COSYSMO tool

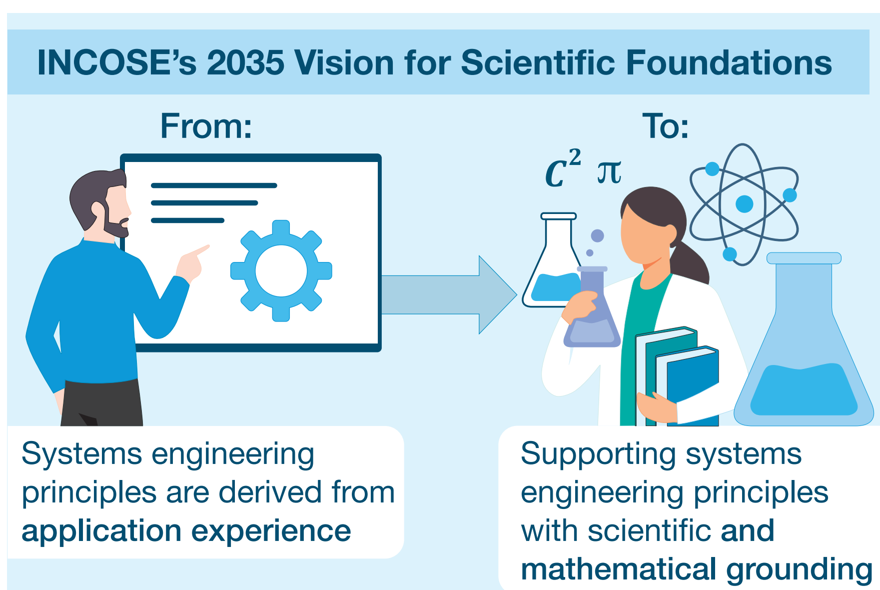


Figure 4. Exploring scientific foundations for systems engineering

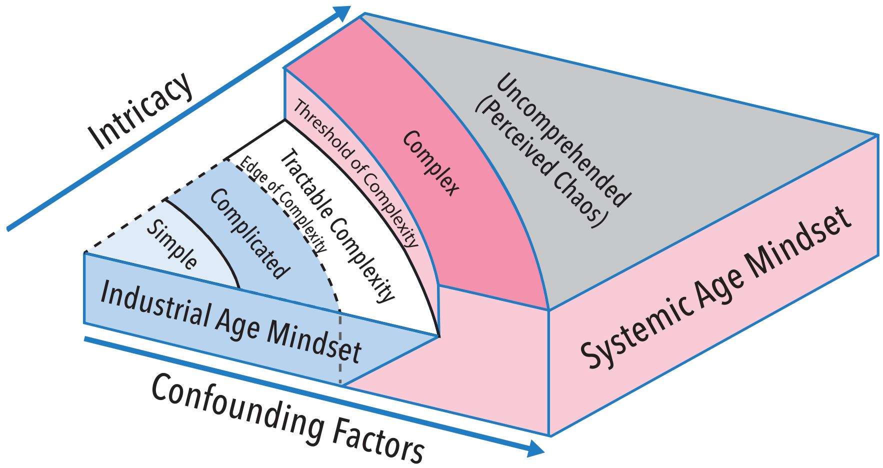


Figure 5. Unified definition of complexity model

it enables the cost of choices in managing complex problems to be considered against the benefits.

The Pleko framework was formed by combining the axis from the unified definition of complexity (UDOC) model, Beale, Dazzi and Tryfonas (2023), in Figure 5 above, with intricacy vs confounding factors (Watson et al. 2019).

4. Creating a Tradecraft Guide

The current systems engineering set of tools understandably reflects past challenges. If you take on board that the world is complex, this will change the way you think, feel, and act" (Boulton et al. 2015). This implies that recognizing complexity means we need to reconsider our practices (methods, approaches, tools, techniques) for success in an uncertain or complex world. More importantly, we also need to change how we see the world, to change our mindset. Together this suggests recognising a challenge is complex is to recognise the need to change everything. Even with this change, however, traditional tools and mindsets continue to be highly valuable for traditional or complicated aspects. What is required is a method of selecting the right tools for the right problem, avoiding the "only using what worked well last time" approach at all costs. The purpose of the tradecraft guide entitled "Making the Improbable Possible" is to help users characterize their problems and select tools suited to them. Though initially being published in physical form as an INCOSE document, the plan is that an online version will be created that will enable a user to quickly identify practices relevant to their needs, and allow the database of practices on how to handle complex problems effectively to expand rapidly, which is necessary in a new emerging area of complex systems engineering. Once this is in place, user feedback can help identify which tools are most effective for what situations, informing the community further. Creating a positive feedback loop on usage will be critical to help us address the most demanding problems to ensure global progress.

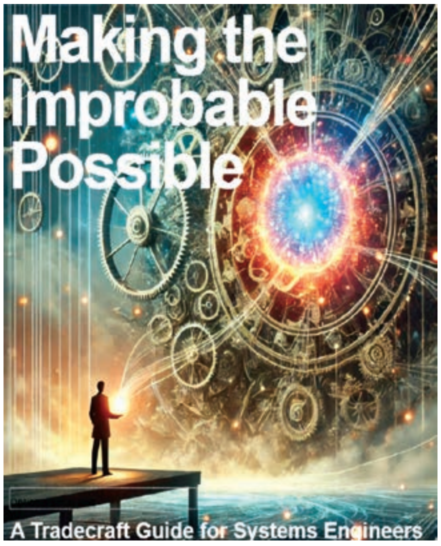


Figure 6. Tradecraft Guide for Systems Engineers (in development)

IMPACT (SO WHAT)

1. Mindset Change:

The INCOSE CSWG recognises that the approach that we need to take to problems with insufficient certainty between cause and effect (complex problems) is fundamentally different to the approach we need to take to problems with sufficient certainty between cause and effect (complicated problems). Even if the terms used to categorise these types of problems change this remains true. As a result, the purpose of this article is to trigger a mindset change, a recognition that we need to do something different to succeed, in what could be called the age of uncertainty, or beyond the Threshold of Complexity. This does not mean throwing away the tools of the past, in actual fact they are more important than ever to minimise the uncertainty as much as possible. But it does include recognising that many, if not most, problems have aspects that are now fundamentally different to those of past requiring change:

1. A different mindset to handle these aspects of new problems
2. Development and utilisation of different approaches and practices for solving these problems
3. The essential importance of characterising and selecting the right approaches for the right problem.

2. Call to Action:

The rise of global uncertainty and associated complexity is occurring at a breath-taking speed and risks widespread systemic failures in many large projects at significant financial, and other costs, to our societies. To address this, we need to change. The CSWG recognises that what we are trying to do is just the beginning of the journey, that many of the thoughts, approaches, and direction are likely to change and improve dramatically as lessons from this new age arise. However, we also recognise that without more community engagement, additional effort and diversity of thinking, we and others working on this will fail to accelerate and communicate the insights required, at the pace required. This could lead to community failure to mitigate the societal systemic failures embedded in applying complicated approaches to complex problems. ■

DISCLAIMERS

During the preparation of this work, an author used ChatGPT 4.0 to do PESTLE (political, economic, social, technological, legal, and environmental) analysis and show outcomes, as well as to create the image under the heading Exploring scientific foundations (Figure 4). The author also used Chat GPT 4.0 and Copilot to reduce duplication of the points raised and word count. The authors take full responsibility for the content of the publication.

REFERENCES

- Baker, S., N. Bloom, and S. J. Davis. n.d. *Economic Policy Uncertainty Index*. Retrieved June 19, 2025 from https://www.policyuncertainty.com/.
- Beale, D., F. Dazzi, and T. Tryfonas. 2024. "Unweaving the Definitions of Complexity." *IEEE Transactions on Systems, Man and Cybernetics: Systems* 54 (2): 682-692, February.

- Beale, D., R. Oosthuizen, D. McKinney, G. Smith, and M. Watson. 2022. "Systems Engineering Heuristics for Complex Systems." 32nd Annual INCOSE International Symposium, Detroit, US-MI: Wiley.
- Beale, D., R. Oosthuizen, D. McKinney, G. Smith, and M. Watson. 2023. "Systems Engineering Heuristics for Complex Systems Revisited." 18th Annual System of Systems Conference, IEEE, Lille, FR.
- Beale, D., A. Pickard, D. McKinney, and R. Valerdi. 2025. "A Systems Engineering Framework for Navigating Complexity." 35th International Council on Systems Engineering Symposium, Ottawa, CA, 26-31 July.
- Bennett, N., and J. Lemoine. 2014. "What VUCA Really Means for You." *Harvard Business Review*, Jan-Feb.
- Boulton, J. G., P. M. Allen, and C. Bowman. 2015. *Embracing Complexity: Strategic perspectives for an age of turbulence*. OUP Oxford.
- INCOSE. 2021a. *Systems Engineering Vision 2035: Engineering Solutions for a Better World*. International Council on Systems Engineering. https://www.incose.org/publications/se-vision-2035.
- INCOSE 2021b. *A Complexity Primer for Systems Engineers, Revision 1*, INCOSE-TP-2021-007-01.
- Morris, L. 2018. *The Big Shift. The 83 most important changes that everyone should know about, and the big shift that changes everything.* Mill Valley, US-CA: Future Lab Press.
- OpenAI. 2025. *ChatGPT response to user query about PESTLE analysis of the Global World Order*. ChatGPT. https://chat.openai.com/.
- Sutherland, J., D. Beale, F. Dazzi, J. Singer, G. Smith, R. Oosthuizen, A. Lanza, K. Cureton, and D. McKinney. 2025. "Managing Variations in Meaning: Guidance for Using 'Complexity' and Related Terms." *Systems Engineering*, submitted for publication.
- Watson, M., R. Anway, D. McKinney, L. Rosser, and J. MacCarthy. 2019. "Appreciative Methods Applied to the Assessment of Complex Systems." INCOSE International Symposium 29 (1): 448-477.

ABOUT THE AUTHORS

Dean Beale has a PhD in organisational complexity, a Master's degree in solid state physics, and is a chartered engineer with the IET. He spent his early career leading R&D at DRA, DERA, and QinetiQ, and was a visiting lecturer at the University of Birmingham. This work naturally led to systems engineering roles and research in organisational complexity at the University of Bristol. He is chair of the International Council On Systems Engineering (INCOSE) Complex Systems Working Group and serves as INCOSE deputy assistant director for international symposium technical operations coordination.

Rudolph Oosthuizen is an associate professor at the graduate school of technology management, University of Pretoria, where he teaches and supervises research in systems engineering and systems thinking. Rudolph has held systems engineering roles in the South African Air Force and Council for Scientific and Industrial Research, contributing to electronic warfare and command and control projects. His academic and professional focus bridges engineering practice, systems thinking, and research to advance the discipline of systems engineering. He is a registered professional engineer with ECSA and a certified systems engineering professional (CSEP) with INCOSE.

Dorothy McKinney is a seasoned systems engineer with nearly four decades of experience in the field. Based in San Jose, California, and originally from New Jersey, she spent 34 years at Lockheed Martin, where she witnessed and contributed to a significant evolution in systems engineering—from a fragmented, bureaucratic approach to a more integrated and adaptive practice. A member of INCOSE since 1992 and an INCOSE fellow, McKinney values the organization for its role in spreading the "systematic application of common sense" that defines systems engineering.

Francesco Dazzi is a senior technologist at the Italian National Institute for Astrophysics (INAF), president 2022-2023 of AISE (INCOSE Chapter Italy), and member of the INCOSE Institute for Technical Leadership. He has more than 20 years of experience mainly dedicated to the construction of ground-based telescopes and technical management of scientific projects. He has a degree in physics, a PhD in mathematics and physics, and more than 250 publications including journal articles and conference proceedings. In 2018 he obtained the CSEP (certified systems engineering professional) certification.

Joshua Sutherland is co-chair of INCOSE's Complex Systems Working Group (CSWG) and co-lead of the Future of Systems Engineering (FuSE) scientific foundations initiative. Joshua is the founder and president of Sutherland Systems Engineering Enterprise (SSEE) Ltd, a consultancy and media group specialising in complex systems and technology development. His team delivers projects for global clients including Ford Motor Company, the European Space Agency, and leading technology firms in Silicon Valley and Japan. Joshua is also a fellow at the University of St. Gallen's Center for Aviation and Space Competence and hosts the world's leading podcast in the systems engineering space.

Understanding Complexity: Defining a Moving Target

Rudolph Oosthuizen, rudolph.oosthuizen@up.ac.za; **Dean Beale,** dean.beale@incose.net; **Francesco Dazzi,** francesco.dazzi.13@gmail.com; **Andrew C Pickard,** andy.pickard@incose.net; **Dorothy McKinney,** dorothy.mckinney@incose.net; **Kenneth Cureton,** kenneth.cureton@incose.net; and **Eileen Patrice Arnold,** eparnold5@aol.com

■ **ABSTRACT**

Systems engineering has transformed its understanding and management of complexity over the past 25 years. This article traces the evolution by analyzing 121 publications from INCOSE's journal and symposium proceedings between 1997 and 2024 to explore how definitions of complexity have shifted from structural metrics to dynamic, context-dependent phenomena shaped by emergent behaviors, stakeholder diversity, and uncertainty. It highlights key distinctions between complicated and complex systems, emphasizing why this matters in design, risk management, and stakeholder engagement. By embracing ambiguity, iterative learning, and multidisciplinary collaboration, engineers can design systems that are resilient and responsive to disruption. The results also underscore the vital role of human factors, including decision-making, cognition, and organizational behavior. We need a mindset shift: success in modern systems engineering depends not on eliminating complexity but on understanding, navigating, and leveraging it to build sustainable, adaptable systems for a changing world.

■ **KEYWORDS:** complexity, systems engineering, emergent behavior, adaptability

INTRODUCTION

In the world of systems engineering, complexity is no longer a buzzword or an occasional challenge. It has become the defining characteristic of modern systems. Whether we are talking about autonomous vehicles, space exploration systems, smart cities, or global health infrastructures, the one constant is complexity. But what do we mean by that term? How has our understanding of it evolved? And why is it so central to the future of systems engineering?

This article seeks to demystify the term "complexity" by tracing its evolution within the systems engineering community over the past 25 years. To do so, the authors reviewed publications from INCOSE's *Systems Engineering* journal and International Symposium proceedings between 1997 and 2024. An initial set of over 4,000 papers was narrowed down using natural language processing and topic modeling, specifically targeting papers where 'complexity' was a prominent theme (Oosthuizen and Pretorius 2020). Manual screening ensured each paper dealt substantively with "complexity," yielding 121 publications for detailed analysis. An analysis of these papers uncovered how complexity's definition, characterization, and methods of addressing have evolved, and why those changes matter.

THE CHALLENGE OF DEFINING COMPLEXITY

Defining complexity has always been difficult, partly because it depends on the context in which the term is used. Early definitions often relied on intuition or were borrowed from other fields, such as philosophy, mathematics, science, or biology. In systems engineering, the difficulty is compounded by the transdisciplinary nature of our work, which spans from technical

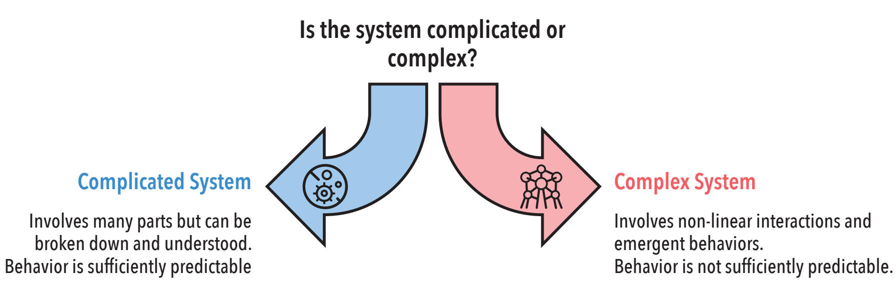


Figure 1. Complicated vs complex system

to human interaction and the influence of the environment. It is also essential to distinguish between complex and complicated systems. This distinction is not just academic but foundational. Although these terms are often used interchangeably, they refer to fundamentally different challenges. The INCOSE Complexity Primer (CSWG 2021) provides a key distinction:

- A complicated system has parts that can be fully understood; hence, their relationships are predictable.
- A complex system, in contrast, involves relationships and behaviors that are not fully comprehensible or predictable due to emergent properties and adaptive behaviors.

Complicated systems may be intricate and involve many parts, but they can be broken down into smaller components and understood through analysis. Their behavior is sufficiently predictable. In contrast, complex systems involve non-linear interactions that often result in emergent behaviors that cannot be easily anticipated. Misunderstanding or overlooking these distinctions can lead to inappropriate system designs, flawed decision-making, and ineffective management strategies. Recognizing whether a system, sub-system, or component is complicated or complex helps engineers choose the right tools and mindsets, influencing everything from risk assessment to stakeholder engagement. This distinction is crucial. Dealing with a system that cannot be fully comprehended may be very different from how you handle a system you can sufficiently comprehend. It allows systems engineers to frame problems in ways that reveal the nature of the challenges involved. It also signals a shift from reductionist approaches to holistic, adaptive thinking, necessitating a new mindset (Snowden and Boone 2007). Also, misunderstanding the difference between complicated and complex problems may lead to project failure (Cavanagh 2013).

A HISTORICAL PERSPECTIVE: 25 YEARS OF EVOLVING DEFINITIONS IN INCOSE

To understand how systems engineering has come to terms with complexity, the authors conducted a detailed review of INCOSE publications between 1997 and 2024. These documents were analyzed using both manual review and advanced natural language processing techniques to extract key themes, definitions, and trends. Here's what we found:

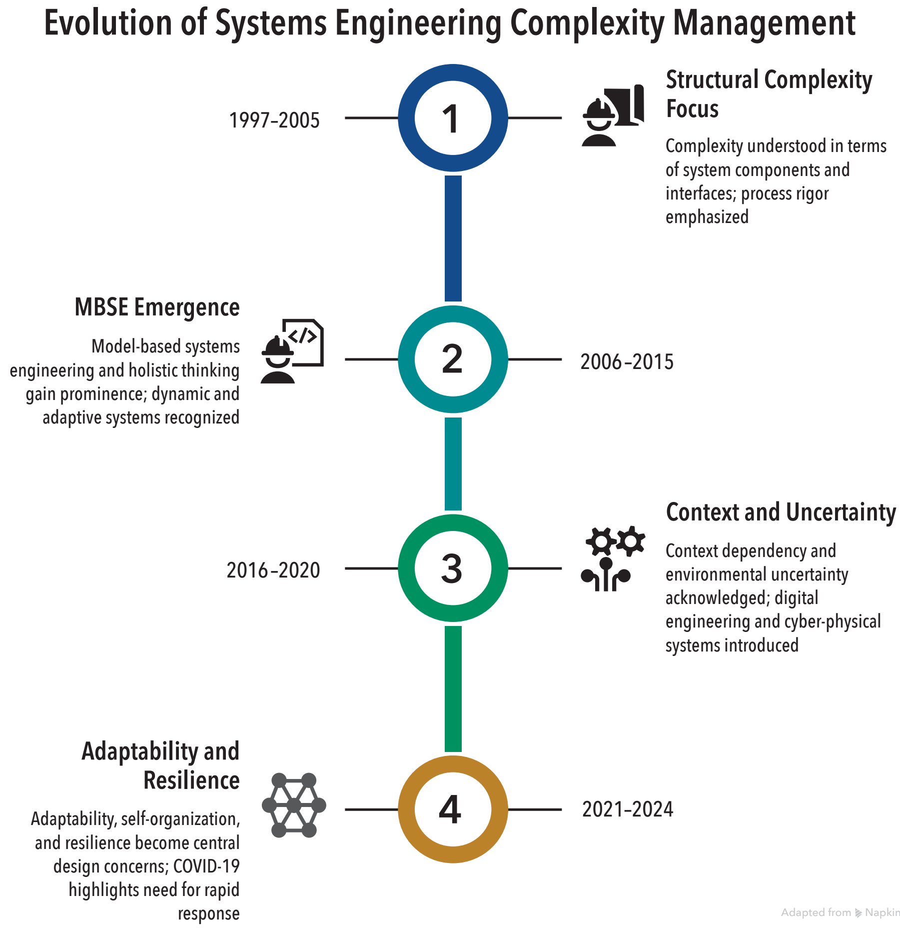


Figure 2. Timeline of complexity understanding in system engineering

1997–2005: Laying the groundwork

In the early years, complexity was understood primarily in structural terms: the number of system components, interfaces, and interdependencies (Magee and de Weck 2004). Complexity was seen as something that could be tamed through process rigor and formal methods (Newbern and Nolte 1999).

2006–2015: Enter Model-based Systems Engineering (MBSE)

This decade saw the emergence of MBSE and a broader appreciation for systems' dynamic and adaptive nature. Complexity began to be framed as something that couldn't always be controlled but needed to be managed through modeling, stakeholder engagement, and holistic thinking (Rouse 2007; Haimes 2012).

2016–2020: Embracing uncertainty and context

This era marked a deeper understanding of how context influences complexity. Engineers recognized that the same system could behave differently in different environments. Digital engineering and cyber-physical systems brought new dimensions to complexity, such as real-time data integration and multi-domain coordination (Keating et al. 2019; Guariniello et al. 2020).

2021–2024: Designing for adaptability and resilience

The most recent years have emphasized adaptability, self-organization, and the limits of traditional approaches. The COVID-19 pandemic underscored the importance of resilience and the ability to rapidly respond to unforeseen disruptions (Liaghati, Mazzuchi, and Sarkani 2021; Haugen et al. 2023).

Figure 2 shows how the understanding of complexity in systems over the past quarter-century has evolved. The systems engineering community has moved from viewing complexity as a structural issue, defined by the number of components or interactions, to recognizing it as a dynamic, context-sensitive phenomenon shaped by stakeholder diversity, environmental uncertainty, and emergent behavior. This evolution, captured through a longitudinal review of INCOSE publications, illustrates a clear trajectory: from control-based strategies to adaptive, resilient, and holistic approaches.

Recognizing and embracing this evolution is critical, reflecting how engineering disciplines evolve from heuristics-based to scientific. If we believe the definition is

SEPTEMBER 2025
VOLUME 28/ ISSUE 4

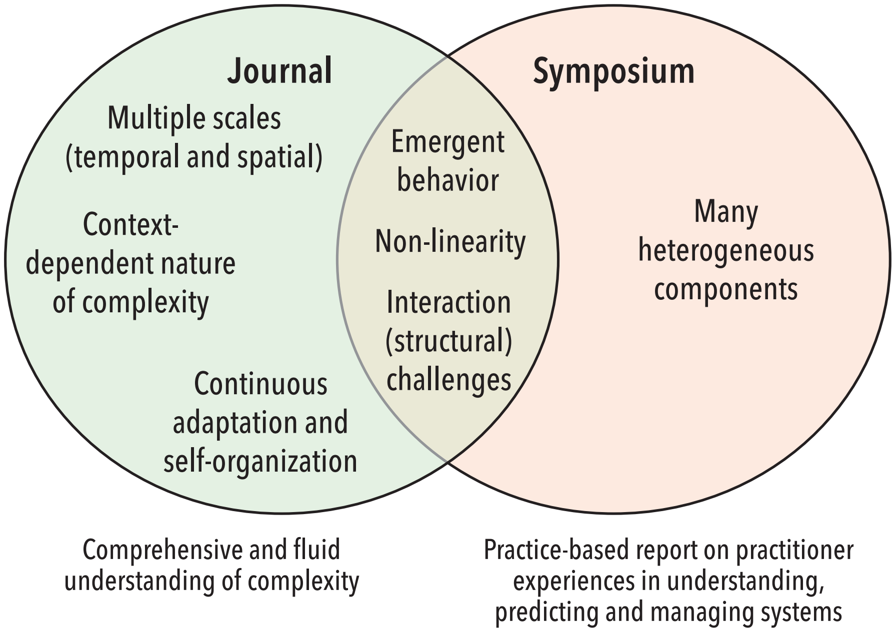


Figure 3. Defining complexity across different sources

static, then the profession cannot progress to develop the scientific foundations it seeks. However, there exists an overlap between the periods, as they are broad approximations. For example, Snowden came out early but became broadly accepted later (Snowden and Boone 2007).

However, this shift also highlights a definitional challenge: systems engineers may justify their understanding of complexity based on the era or source they reference. Due to the evolution of the definition of complexity, it is easy for an author or community to choose a definition that matches their preferences at some point on the evolutionary path. If the definition is not clearly stated, ideally in the context of other definitions, this approach can lead to the inability of the term to evolve to what it needs to be to enable the profession to progress. The term must be defined and justified to help the essential evolution. Even if we use the exact etymology of the term complexity, we may have struggled with different causes of complexity in the past, i.e., the high number of system relationships. Nowadays, it may be systems' adaptability, and it might depend on something different in the future.

Uncertainty can persist without anchoring definitions in recent, peer-reviewed literature about the current context. It is critical to ground discussions of complexity in contemporary scholarship that reflect the current, nuanced understanding of systems as evolving, context-dependent entities. Understanding this definitional evolution is vital, as it reflects the growing sophistication in how engineers perceive, model, and manage the inherently unpredictable nature of modern systems (Beale, Dazzi, and Tryfonas 2023; CSWG 2021).

In addition, it is important to recognize the definition perspective can change based on context. An analysis of the papers shows that the journal and conference papers provide different perspectives on the definition in general as shown in Figure 3 reflecting the audience and community from which they evolved. Demonstrating that the symposium papers tend to emphasize the number of parts, in contrast to journals which focus on more dynamic aspects.

WHAT MAKES A SYSTEM COMPLEX?

While systems with more diverse components exhibit higher complexity, it is essential to recognize that increased scale and scope do not inherently result in a complex system. Instead, they elevate the opportunity or probability for complexity to emerge. This distinction is essential, as scale and component count remain among the most debated dimensions in defining complexity. A system can be large yet remain complicated rather than complex if its behavior is predictable and linear. Conversely, even smaller systems can exhibit true complexity when non-linear interactions, emergence, and dynamic context play significant roles. Therefore, while scale and scope are influential, they should not be viewed as sufficient conditions for complexity but as contributing factors that, combined with interaction patterns, environmental uncertainty, and stakeholder diversity, may lead to complex behavior (Cilliers 2002).

When considering uncertainty, unpredictability and levels of understanding, it is clear that they reflect different but interconnected aspects of uncertainty

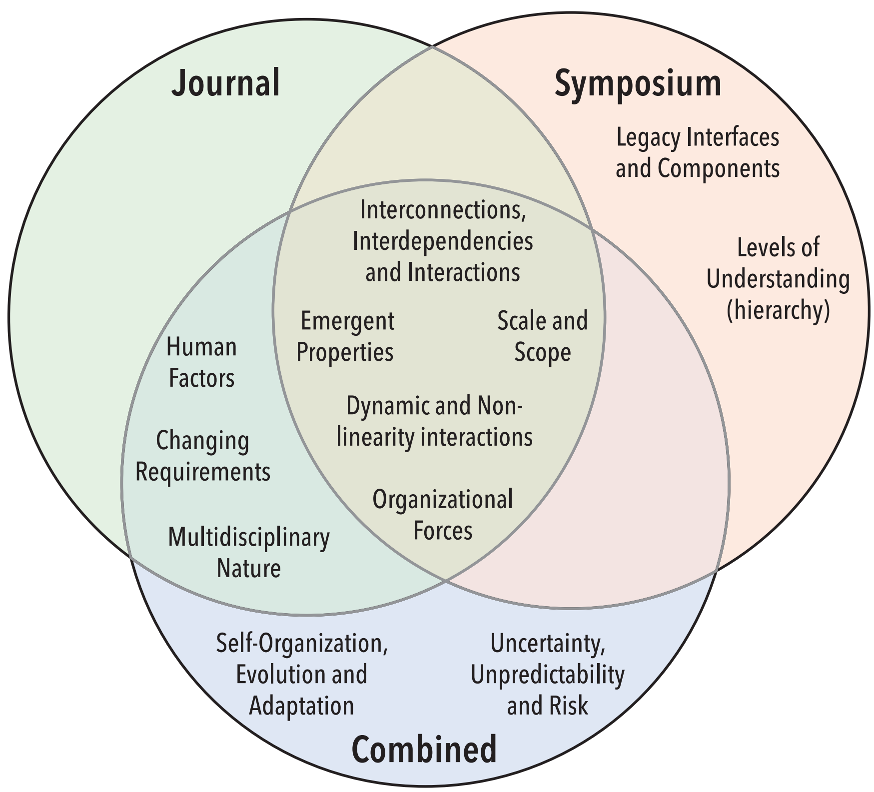


Figure 4. Twelve (12) key elements that consistently drive (cause or amplify) complexity

Table 1. Characteristics and focus of complexity in the different eras

	1997–2005	2006–2015	2016–2020	2021–2024
Causes of Complexity	Increased system requirements. Interdisciplinary integration and interdependencies. Non-linearity and unpredictability in system interactions.	Sociotechnical integration. Globalization of development teams. Increase in software-intensive systems.	Cyber-physical systems and IoT. Increasing system autonomy and intelligence. Accelerated pace of technology integration.	Growing reliance on AI and autonomy. Systems-of-systems with emergent behaviors. System evolution during operation.
Observed Effects	Difficulty in requirement management and integration. Reduced traceability of decisions. Increased likelihood of emergent behaviors.	Misalignment between design and user needs. Integration delays and rising costs. Design changes due to misunderstood requirements.	Difficulty in validating emergent behavior. Data overload and inconsistency. Challenges in lifecycle synchronization across disciplines.	Difficulty in predicting long-term performance and safety. Uncertainty in human-machine interaction. Limited assurance and trustworthiness.
Systems Engineering Tools	Traditional modelling methods. Early architecture frameworks. Structured systems engineering lifecycle processes.	MBSE with SysML. Architecture frameworks (DoDAF, TOGAF). Simulation tools.	Advanced MBSE with digital threads. Integration of ontologies and semantic models. Simulation-based design and V&V.	Digital twins and integrated digital threads. AI-supported tools and smart models. MBSE with agile systems engineering and DevSecOps.
Key Characteristics	Structural (size and integration). Behavioral unpredictability due to coupling. Increased stakeholder coordination needs.	Dynamic and evolving SoS. Interaction complexity. Stakeholder heterogeneity complicates requirements.	Adaptive and self-configuring systems. Network-centric characteristics. Openness and distributed development.	Runtime adaptability and continuous system evolution. Unpredictability and non-determinism in behavior. Complex human-system interaction.
Focus of Systems Engineering	Standards to manage system requirements and lifecycle (risk and uncertainty) through documentation and processes.	MBSE to manage traceability and integration. Stakeholder involvement and validation. Emergent behavior and adaptability.	Continuous engineering and integration. Adaptability, interoperability, and resilience. Automated decision-support and validation.	Trust, assurance, and resilience. Continuous V&V and operational feedback. Collaborative systems engineering with real-time adaptation.

in complex systems. Unpredictability and risk primarily address future-state uncertainty, where the inability to foresee outcomes stems from emergent behaviors and dynamic interdependencies. In contrast, "levels of understanding" relates to now-state uncertainty, highlighting the challenge of interpreting current system behavior across hierarchical levels. Recognizing that both dimensions represent critical facets of uncertainty is essential. Too often, practitioners focus on one while neglecting the other, leading to incomplete assessments and suboptimal strategies. A holistic approach requires acknowledging and managing both present ambiguity and future unpredictability (Paté-Cornell 2012; Beale, Tryfonas, and Young 2017).

Understanding these elements helps engineers diagnose complexity and decide whether a system is complicated or complex. More importantly, this nuanced understanding reshapes how systems engineering is practiced today. Modern systems engineers must now account for technical specifications and how systems behave in varied, often unpredictable contexts. Emergent behavior, non-linear dynamics, and organizational influences necessitate a shift from linear planning to iterative learning and adaptation. Traditional reductionist methods are increasingly supplemented, or even replaced, by holistic approaches emphasizing resilience, agility, and whole-system awareness. As a result, the identification and acknowledgement of these complexity drivers are now integral to designing systems that are robust, sustainable, and capable of thriving amid change.

From the 121 papers, we identified 12 key elements that consistently cause or amplify complexity. The difference in focus between the sources stems from the distinct purposes of journals and conferences, as seen in Figure 4. The topics in journals often involve subjective interpretation of context-specific applications or require a nuanced discussion that fits the extended format of journal articles. The conference papers highlight ongoing research or preliminary findings, making them ideal for exploring concepts where peer input and feedback are critical. The journals tend to emphasize lasting, theoretical contributions, while conferences prioritize the immediate exchange of ideas and practical insights. Therefore, the combined set caters to elements of complexity that are universally important, bridging theory and practice. Understanding these differences helps in tailoring the dissemination of research and the engagement with appropriate audiences. No single element alone creates complexity— the interaction and accumulation of these factors do.

The longitudinal analysis also identified several recurring characteristics contributing to a system's complexity, as seen in Table 1. An interesting observation is that the systems engineering effort (focus) and tools aim to solve the problems experienced in the preceding period. This evolution contrasts significantly with earlier, more reductionist systems engineering approaches focused primarily on decomposing problems and optimizing individual components in isolation (Kossiakoff et al. 2011). Today, the influence of complexity drivers compels engineers to adopt more flexible, integrative methodologies that account for both technical and socio-organizational variables. References such as Jackson (2019) and Remington and Pollack (2016) further reinforce this shift, illustrating that different system contexts require different decision-making approaches. Likewise, Beale, Dazzi, and Tryfonas (2023) highlight the importance of precise language and clarity in distinguishing between types of complexity, directly impacting the efficacy of engineering strategies and outcomes.

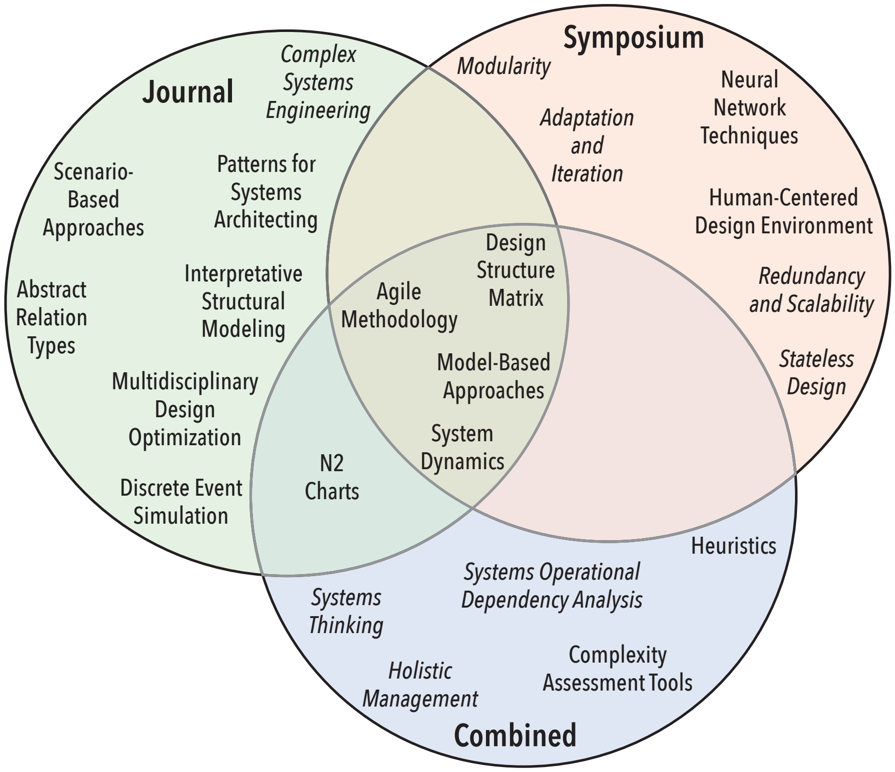


Figure 5. Tools and methods to address complexity. "Combined" tools are those that bridge the gap between theory and practice for broad applicability. (Strategies are highlighted in italics; tools and methods are not in italics.)

FROM CONTROL TO INFLUENCE: TOOLS FOR MANAGING COMPLEXITY

As seen in Table 1, the traditional engineering mindset aims to control and predict. However, as systems become more complex, that mindset must shift. Instead of control, we strive for influence (Meadows 2009). Instead of prediction, we seek resilience. The shift from control to influence carries significant implications for today's systems engineers. Engineers must now design with uncertainty, evolution, and stakeholder diversity in mind rather than assuming that a system's behavior can be fully anticipated and dictated through deterministic models. This necessitates a departure from rigid, perfection-driven design processes toward more flexible, iterative development cycles with living systems where we experiment and learn to increase understanding of the system and its behavior (Jackson 2019). It is about accepting that there is no complete control and knowledge, and adapting to the system's outcomes to continue the learning journey.

Embracing such flexibility reduces the pressure for perfection upfront. This facilitates faster decision-making and system realization and empowers engineers to respond more effectively to the accelerating pace of technological and contextual change while avoiding analysis paralysis. Consequently, systems engineering is becoming more agile, enabling real-time learning and refinement. Engineers are also called to develop stronger skills in communication, facilitation, and systems thinking— disciplines that help bridge gaps between technical solutions and human needs. Furthermore, the rising emphasis on holistic and collaborative management underscores a shift from siloed expertise toward multidisciplinary approaches reflecting modern sociotechnical systems' interconnected nature.

The tools and frameworks outlined above highlight a key implication for today's systems engineers: success now depends on technical proficiency, strategic flexibility, and systems awareness. Navigating complexity requires practitioners to be comfortable with ambiguity, capable of working across disciplines, and prepared to iterate on solutions as systems evolve. This shift also demands that engineers cultivate broader competencies, including stakeholder engagement, ethical foresight, and framing problems within the larger sociotechnical context. Ultimately, managing complexity is less about eliminating uncertainty and more about building systems — and teams — that can adapt, learn, and respond effectively to the unexpected.

Managing complexity requires a layered approach combining both strategies and technical tools. As seen in Figure 5, the strategies are highlighted in italics; the rest are the technical tools. Journals and symposiums play complementary roles in advancing the use of tools for managing complexity. The symposiums highlight modularity, adaptation and iteration, and stateless design, which are practical tools directly relevant to practitioners refining or implementing systems. Journals lay the theoretical groundwork for robust methods, while the symposiums focus on practical applications and real-world challenges. The combined list emphasizes universal principles like systems thinking and holistic management, which appeal to theoretical and practical audiences. The tools derived from the combined dataset bridge the gap between theory and practice for broad applicability. The diversity of these tools reflects the need to tailor interventions to each system's complexity profile. The listed complexity tools are also often part of different frameworks for addressing complexity in systems.

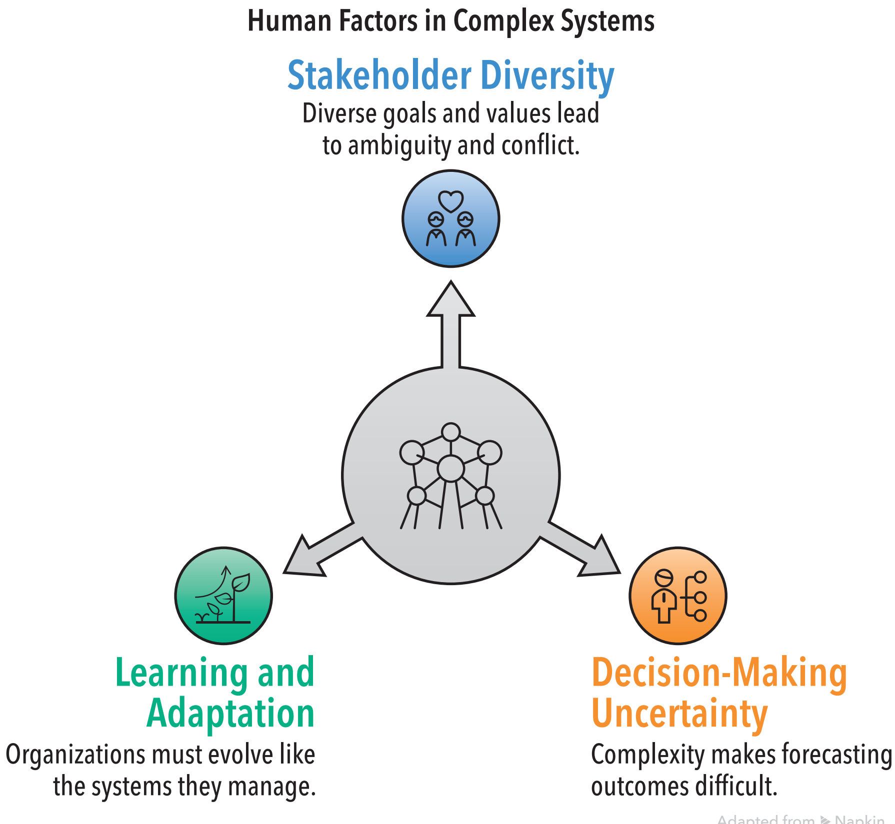


Figure 6. Human factors in complexity

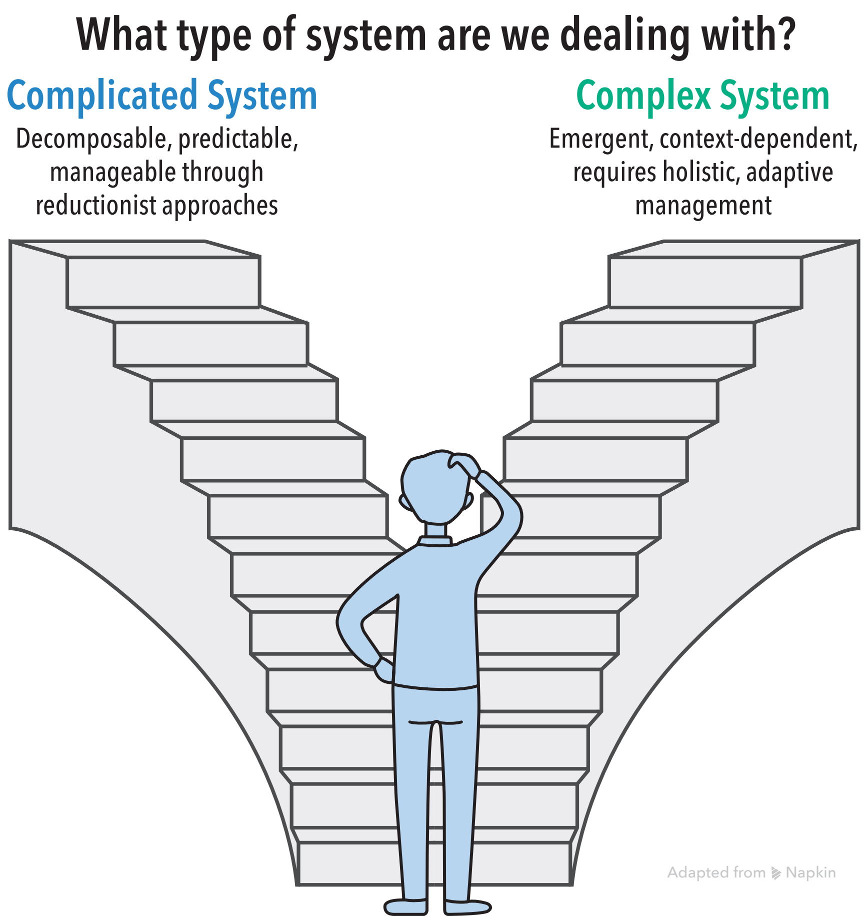


Figure 7. What type of system are we dealing with

THE HUMAN SIDE OF COMPLEXITY

While technology often gets the spotlight, the human system element is just as critical and needs to be as resilient. People design, operate, and interpret systems. As systems become more complex, human factors such as communication, cognition, and organizational behavior become central.

- **Stakeholder Diversity:** Different goals and values introduce ambiguity and conflict.
- **Decision-Making Under Uncertainty:** Complexity makes it harder to forecast outcomes.
- **Learning and Adaptation:** Organizations must learn and evolve like the systems they manage.

The research underscores the need for transdisciplinary approaches that foster collaboration, drawing insights from psychology, sociology, and organizational science.

CONCLUSION

The distinction between complicated and complex systems as defined by INCOSE (CSWG 2021) is not just academic but foundational. Although these terms are often used interchangeably, they refer to fundamentally different challenges. Complicated systems may be intricate and involve many parts, but they can be broken down into smaller components and understood through analysis. Their behavior is generally predictable. In contrast, complex systems involve non-linear interactions that often result in emergent behaviors that cannot be easily anticipated. Misunderstanding or overlooking these distinctions can lead to inappropriate system designs, flawed decision-making, and ineffective management strategies. Recognizing whether a system is complicated or complex helps engineers choose the right tools and mindsets, influencing everything from risk assessment to stakeholder engagement. It has profound implications for designing, testing, and operating systems. Complicated systems are manageable through reductionist approaches. Complex systems require holistic, adaptive management. Understanding which type of system you're working with can inform everything from stakeholder engagement to risk assessment. If these terms were to continue to evolve, to mean something different to how INCOSE has defined it, then alternative terms would need to be created to represent the necessary separation between these distinct types of challenges.

As systems engineering continues to mature in its treatment of complexity, several areas warrant focused research and community engagement. A key challenge is defining and operationalizing the boundary between complicated and complex systems, particularly around what might be termed resolvable complexity. This includes exploring the thresholds at which systems transition from being analytically tractable to requiring adaptive, non-linear strategies. The need for deeper theoretical and empirical work on emergence is closely tied to this, specifically, whether and how emergent behaviors arise in complicated systems and what mechanisms distinguish emergent phenomena in complex versus resolvable systems. Emergent behaviors raise concerns about predictability versus adaptability— a tension increasingly relevant with the rise of AI-enabled systems. Addressing how safety-critical systems can be both certifiable and adaptable under conditions of complexity is vital. These areas highlight the need for academic inquiry and collaborative dialogue between engineers, regulators, and stakeholders. Research should advance conceptual clarity, develop practical tools, and cultivate inclusive forums to navigate these unresolved and emerging challenges. ■

DISCLAIMER

During the preparation of this work, an author used ChatGPT 4.o to summarise articles as part of the literature review. The authors also used Chat GPT 4.o and Grammarly to improve grammar, readability, and reduce duplication and count. Some of the images were created with napkin.ai. The authors take full responsibility for the content of the publication.

REFERENCES

- Beale, Dean, Francesco Dazzi, and Theo Tryfonas. 2023. “Unweaving the Definitions of Complexity.” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.
- Beale, Dean, Theo Tryfonas, and Michael Young. 2017. “Evaluating Approaches for the next Generation of Difficulty and Complexity Assessment Tools.” In *2017 IEEE Technology & Engineering Management Conference* (TEMSCON), 227–33. IEEE.
- Cavanagh, Michael. 2013. “Project Complexity Assessment.” *ICCPM, Kindle*.
- Cilliers, Paul. 2002. *Complexity and Postmodernism: Understanding Complex Systems*. Routledge.
- CSWG. 2021. “A Complexity Primer for Systems Engineers.” https://www.incose.org/docs/default-source/ProductsPublications/a-complexity-primer-for-systems-engineers.pdf.
- Guariniello, Cesare, Ali Khalid Raz, Zhemei Fang, and Daniel DeLaurentis. 2020. “System-of-systems Tools and Techniques for the Analysis of Cyber-physical Systems.” *Systems Engineering* 23 (4): 480–91.
- Haimes, Yacov Y. 2012. “Modeling Complex Systems of Systems with Phantom System Models.” *Systems Engineering* 15 (3): 333–46.
- Haugen, Rune Andre, Nils-Olav Skeie, Gerrit Muller, and Elisabet Syverud. 2023. “Detecting Emergence in Engineered Systems: A Literature Review and Synthesis Approach.” *Systems Engineering* 26 (4): 463–81.
- Jackson, Michael C. 2019. *Critical Systems Thinking and the Management of Complexity*. John Wiley & Sons.
- Keating, Charles B, Polinpapilinho F Katina, Ra’ed Jaradat, Joseph M Bradley, and Richard Hodge. 2019. “Framework for Improving Complex System Performance.” In *INCOSE International Symposium*, 29:1218–32. Wiley Online Library.
- Kossiakoff, Alexander, William N Sweet, Samuel J Seymour, and Steven M Biemer. 2011. *Systems Engineering Principles and Practice*. 2nd edition. John Wiley & Sons.
- Liaghati, Hassan Robert, Thomas A Mazzuchi, and Shahram Sarkani. 2021. “Utilizing a Maximin Optimization Approach to Maximize System Resiliency.” *Systems Engineering* 24 (3): 144–57.
- Magee, Christopher L, and Oliver L. de Weck. 2004. “3.13 Complex System Classification.” In *INCOSE International Symposium*, 14:471–88. Wiley Online Library.
- Meadows, Donella H. and Diana Wright. 2009. *Thinking in Systems. ROUTLEDGE Limited*.
- Newbern, David, and Jerry Nolte. 1999. “Engineering of Complex Systems: Understanding the Art Side.” *Systems Engineering* 2 (3): 181–86.
- Oosthuizen, Rudolph, and Leon Pretorius. 2021. “Analysis of INCOSE Systems Engineering Journal and International Symposium Research Topic Trends.” *Systems Engineering* 24 (4): 203–220.
- Paté-Cornell, Elisabeth. 2012. “On ‘Black Swans’ and ‘Perfect Storms’: Risk Analysis and Management When Statistics Are Not Enough.” *Risk Analysis: An International Journal* 32 (11): 1823–33.
- Remington, Kaye, and Julien Pollack. 2016. *Tools for Complex Projects*. Routledge.
- Rouse, William B. 2007. “Complex Engineered, Organizational and Natural Systems: Issues Underlying the Complexity of Systems and Fundamental Research Needed to Address These Issues.” *Systems Engineering* 10 (3): 260–71.
- Snowden, David J, and Mary E Boone. 2007. “A Leader’s Framework for Decision Making.” *Harvard Business Review* 85 (11): 68.

ABOUT THE AUTHORS

Rudolph Oosthuizen is an associate professor at the graduate school of technology management, University of Pretoria, where he teaches and supervises research in systems engineering and systems thinking. Rudolph has held systems engineering roles in the South African Air Force and Council for Scientific and Industrial Research, contributing to electronic warfare and command and control projects. His academic and professional focus bridges engineering practice, systems thinking, and research to advance the discipline of systems engineering. He is a registered professional engineer with ECSA and a certified systems engineering professional (CSEP) with INCOSE.

Dean Beale has a PhD in organisational complexity, a Master’s degree in solid state physics, and is a chartered engineer with the IET. He spent his early career leading R&D at DRA, DERA, and QinetiQ, and was a visiting lecturer at the University of Birmingham. This work naturally led to systems engineering roles and research in organisational complexity at the University of Bristol. He is chair of the International Council On Systems Engineering (INCOSE) Complex Systems Working Group and serves as INCOSE deputy assistant director for international symposium technical operations coordination.

Francesco Dazzi is a senior technologist at the Italian National Institute for Astrophysics (INAF), president 2022-2023 of AISE (INCOSE Chapter Italy), and member of the INCOSE Institute for Technical Leadership. He has more than 20 years of experience mainly dedicated to the construction of ground-based telescopes and technical management of scientific projects. He has a degree in physics, a PhD in mathematics and physics, and more than 250 publications including journal articles and conference proceedings. In 2018 he obtained the CSEP (certified systems engineering professional) certification.

Andrew C. Pickard joined Rolls-Royce in 1977 after completing a PhD at Cambridge University in fatigue and fracture of metals and alloys. He is a Rolls-Royce associate fellow in system engineering; a fellow of SAE International; a fellow of the Institute of Materials, Minerals and Mining; a chartered engineer; and a member of INCOSE. He was chair of the SAE Aerospace Council from 2015 to 2019 and INCOSE chief of staff from the 2016 International Workshop to the 2024 International Workshop. He retired from Rolls-Royce Corporation at the end of 2022. He is currently INCOSE assistant director for International Symposium Technical Operations Coordination and co-chair of the INCOSE Complex Systems Working Group.

Dorothy McKinney is a seasoned systems engineer with nearly four decades of experience in the field. Based in San Jose, California, and originally from New Jersey, she spent 34 years at Lockheed Martin, where she witnessed and contributed to a significant evolution in systems engineering—from a fragmented, bureaucratic approach to a more integrated and adaptive practice. A member of INCOSE since 1992 and an INCOSE fellow, McKinney values the organization for its role in spreading the "systematic application of common sense" that defines systems engineering.

Kenneth L. Cureton holds a BS in physics from California State University Los Angeles and an MS in systems architecting and engineering from USC, where he has served as a part-time lecturer since 1996. He retired from Boeing (Rockwell) after 29 years, with additional experience spanning 16 years in commercial and government sectors. His expertise includes enterprise systems architecting, system-of-systems modeling and integration, complex networked systems interoperability, resilience engineering, and avionics design for manned and unmanned space systems. Cureton chairs the INCOSE Resilient Systems Working Group and serves on AIAA Space Settlement and Systems Engineering Technical Committees. He previously co-chaired the IEEE SMC MBSE Working Group and served as technical council chair emeritus for the NCO Industry Consortium.

Eileen Patrice Arnold, fellow of INCOSE, INCOSE expert systems engineering professional (ESEP), past project management professional (PMP), and a recipient of the MFESTS Charles W. Britzius Distinguished Engineer Award has enjoyed and thrived for almost 40 years as an aircraft electrical systems engineer, weapons systems engineer, instructor and mentor to other engineers and managers. In the mid-1990's she discovered her passion for systems engineering. She is and has been an active INCOSE volunteer and author since 1996. Although retired, she continues to be active in the INCOSE working groups associated with adaptable, resilient, and complex systems.

SEPTEMBER 2025
VOLUME 28/ ISSUE 4

Systemic Elegance: Clarifying Complexity and Emergence for Engineering Practice

Gary Smith, grs0036@gmail.com

ABSTRACT
In systems engineering, elegance is often associated with simplicity and control, but this risks ignoring the deeper systemic nature of elegance as coherence between complexity and purpose. This article reframes elegance not as minimalism, but as the systemic sufficiency of structure, behavior, and context. Drawing from systems science foundations and the triad of fit–form–function, we argue that true systemic elegance arises when relational complexity is harnessed, not suppressed, to yield emergent coherence across levels. Elegant systems are those that integrate complexity without introducing unnecessary complication; they achieve just enough richness to engage with the variety of their environment, while avoiding overdesign. This balance is not accidental; it results from rigorous architecting and purposeful design. Using examples and distinctions, this article offers a framework for systems engineers to recognize, cultivate, and evaluate elegance as a dynamic outcome of systemic coherence.

INTRODUCTION: SEEKING COHERENCE IN A COMPLEX WORLD

Modern systems engineering is increasingly challenged by the scale, speed, and interdependence of change. In response, complexity has become a central concern, but also a source of confusion. It is often treated as a synonym for difficulty, unpredictability, or technical overload. Yet complexity, properly understood, is neither inherently negative nor synonymous with complication (Mobus and Kalton 2014, Simon 2019) It is a fundamental property of systems: an expression of relational richness, arising from the meaningful interactions among a system's parts and with its environment (Von Bertalanffy 1968). The term "elegance" is sometimes used in contrast with complexity, as if it implies simplicity or aesthetic minimalism. But from a systems perspective, elegance refers not to the absence of complexity, but to its meaningful alignment. A system is elegant when it exhibits just enough complexity to achieve coherence across scale, context, and purpose (Mobus and Kalton 2014). This article reframes complexity and emergence in terms of fit–form–function (FFF), a long-standing engineering triad that, when interpreted systemically, provides a powerful diagnostic and generative lens. We build on the INCOSE definition of a system as "an arrangement of parts or elements that together exhibit behavior or meaning that the individual constituents do not" (Sillitto et al. 2019), recognizing that such properties emerge not only from internal structure but also from dynamic relationships with the external environment. To practice systemic elegance is to cultivate this dynamic alignment intentionally. It is to move from managing difficulty to designing for coherence. In the sections that follow, we explore how this shift enables a richer understanding of complexity, a more grounded view of emergence, and a principled foundation for engineering systems that are both adaptive and meaningful.

BEYOND COMPLEXITY AND SIMPLICITY: TOWARD SYSTEMIC ELEGANCE

The term complexity is frequently conflated with difficulty or uncertainty, but in systems science it refers to a more precise quality: the richness of dynamic relationships within and beyond a system. Simon (2019) expresses that "*The inner environment, the hardware, is simple, Complexity emerges from the richness of the outer environment, both the world apprehended through the senses and the information about the world stored in long-*

term memory." Simon (2019) also reflects, "*Information theory explains organized complexity in terms of the reduction of entropy (disorder) that is achieved when systems (organisms, for example) absorb energy from external sources and convert it to pattern or structure.*" Both of these perspectives express that a complex system is not simply intricate; it is one whose behavior emerges from nontrivial interactions among its components in relation to its environment, often in unpredictable ways (Anderson 1972, Simon 2019). In this framing, complexity is not a failure state, it is a characteristic of life, cognition, and adaptive function. Biological organisms, ecologies, social institutions, and engineered systems all display complexity, not as noise or inefficiency, but as the substrate of responsiveness, resilience, and learning (Mobus and Kalton 2014, Von Bertalanffy 1968). The challenge arises not from the presence of complexity, but from misalignment between the system and its context. A system may become unstable or brittle when its internal variety (Ashby 1961) cannot keep pace with environmental demands, or when it generates unintended side effects. Elegance, in this context, is not the elimination of complexity, but its purposeful shaping, where relationships, variety, and structure are aligned to support systemic coherence. This perspective invites a reframing of the engineering task: from minimizing complexity to working with it consciously, understanding it as a medium through which emergence becomes possible, and intelligence becomes distributed. Complexity is the soil from which coherence can grow, if the conditions of alignment are met.

UNDERSTANDING EMERGENCE: ALIGNMENT, NOT ACCIDENT

Emergence refers to the appearance of new properties, behaviors, or meanings at the level of the system that are not reducible to its parts. These properties arise from the interactions among components and with the system's environment (Anderson 1972, Von Bertalanffy 1968). Emergence is not a mysterious byproduct, it is a systemic phenomenon, often lawful, patterned, and diagnosable (Mobus and Kalton 2014). In engineering, emergent properties may include functional capabilities, performance characteristics, or system-level behaviors that result from integration. But emergence is not merely additive, it is relational. What matters is not only the parts, but how they relate, and how the system as a whole interacts with external systems. A change in the environment may reveal or suppress emergence, just as internal reconfiguration can (Mobus and Kalton 2014) .This dynamic is reflected in the INCOSE definition of a system (Sillitto et al. 2019), which includes the potential for emergent behavior or meaning. It is also consistent with Ashby's Law of Requisite Variety (Ashby 1961): systems must maintain sufficient internal diversity to respond to environmental complexity. In systems terms, meaning is not confined to human interpretation. All systems derive meaning through their interactions with their environment, by responding to signals, exchanging matter, energy, or information, and adapting in relation to purpose or constraint. Meaning arises when the system's internal organization becomes aligned with, or responsive to, patterns in its external context. This applies equally to biological organisms, artificial agents, and engineered systems. Emergence, then, can be seen as a form of coherence, when the system expresses a capability that is meaningful in its context, adaptive in its function, and structured in a way that is not preprogrammed but arises through configuration and relationship. Such emergence may be desirable, as in the case of intelligence, adaptability, or resilience, but it can also be pathological, leading to runaway feedback, coordination breakdowns, or unexpected harm. Understanding emergence as a diagnostic expression of FFF alignment allows engineers to engage with it not as a surprise, but as a traceable outcome of systemic design.

FIT–FORM–FUNCTION: A HOLARCHIC TRIAD FOR ELEGANCE

The triad of FFF is a set of long-standing systems engineering concepts, rooted in defense logistics and configuration management (Handbook 2001). Together they provide the framing for heuristics (principles based on experience) how components must align with their environment while maintaining interchangeability. The concepts and associated principles have been formalized in US defense procurement policy since the 1960s. The concept of FFF, has also been articulated in design literature by Dumas and Mintzberg (1991), who explored how these three dimensions interact in organizational and design systems, reinforcing its value as a systemic integrative principle.

However, when viewed through the lens of systems science, FFF becomes a powerful diagnostic model that aligns with the concepts of holarchy, emergence, and complexity (Mobus and Kalton 2014).

- Form refers to the system's structure, its topology, interfaces, architecture, and configuration.
- Function refers to what the system does, its performance, operations, and purpose as realized in action.
- Fit refers to how well the system relates to its environment, its alignment with contextual needs, constraints, and meaning.

FFF does not simply describe three attributes, it captures a systemic dynamic. Form enables Function, which expresses Fit. A misalignment in any one dimension may result in incoherence, brittleness, or unintended outcomes (Troncale 2013). This triad offers a practical way to make elegance visible: not as simplicity, but as sufficiency and coherence across these dimensions. Moreover, FFF reflects a holarchic structure, systems within systems, each simultaneously a part and a whole (Koestler 1968, Von Bertalanffy 1968). As Anderson (1972) famously noted, "*more is different*," complex systems exhibit emergent properties that require new principles of understanding and cannot be fully explained by reduction to their parts. This reinforces the view that elegance is not about reduction, but about relational clarity. Each level of system, from component to capability, expresses a coherent alignment of FFF appropriate to its scope. This framing also provides a valuable lens across the system lifecycle, supporting more meaningful validation and verification (V&V), architectural coherence reviews, and cross-disciplinary alignment. Rather than assessing parts in isolation, engineers can assess whether each level of the system exhibits coherence across context (Fit), structure (Form), and performance (Function).

Consider, for example, the transformation of a global defence and security enterprise from document-centric engineering to a fully digital, model-based business management system. This shift is not just about digitizing artifacts, it is about architecting an intelligent enterprise system. The elegance lies in the systemic alignment of Form (integrated platforms, modular data structures, digital thread), Function (engineering workflows, governance processes, operational decision-making), and Fit (strategic needs, regulatory environments, customer missions). As these dimensions converge, augmented intelligence emerges, not as a standalone technology, but as a system property: the ability of the enterprise to perceive patterns, learn from operations, and adapt continuously. This intelligence arises through coherence, when human roles, digital infrastructure, and process architectures act in unison. It enables the enterprise not only to manage complexity, but to evolve with it, revealing a deeper form of elegance, rooted in context-aware sufficiency and reflective capability.

ARCHITECTING AND DESIGNING: TWO VIEWS IN HOLARCHIC RELATION

In systems practice, architecting and designing are often presented as distinct phases or roles. But this distinction is not fixed, it is relational. Both engage deeply with FFF, but from different vantage points in a holarchy, a nested structure where every system is both a part and a whole (Koestler 1968, Von Bertalanffy 1968).

Architecting is the act of engaging with the system as a whole in relation to its environment or containing system. It frames the systems:

- Purpose – Why does this system exist?
- Role – What contribution must it make?
- Identity and Boundaries – What differentiates it, and how does it interact?
- Function – What outcomes must it produce externally?

The architect's concern lies with ensuring FFF at the level of the whole and shaping a vision of the solution that enables coherence. Architecting is thus a *context-facing activity*: it interprets what is needed for the system to remain sufficient and adaptable within a broader ecosystem (Ashby 1961, Mobus and Kalton 2014).

This orientation is exemplified in the digital transformation of a business management system (BMS). Here, architecting involves reimagining the enterprise not as a collection of functions, but as a dynamic system of systems, spanning engineering, operations, and governance, and capable of strategic coherence and agility across geopolitical, regulatory, and mission contexts.

Designing, by contrast, views the system as a whole-to-be-composed, working from the parts upward to realize the architected intent. Designers ask:

- How can the necessary functions be enacted through subsystems?
- How must interfaces, flows, and interactions be shaped?
- What structural, material, or logical choices ensure integrity and performance?

Designing is a composition-focused activity. It realizes FFF through the coordinated development of parts, ensuring that internal complexity is purposeful and sufficient, not excessive (Ashby 1961). As Herbert Simon observed, "several components in any complex system will perform particular sub functions that contribute to the overall function… to design such a complex structure, one powerful technique is to discover viable ways of decomposing it into semi-independent components corresponding to its many functional parts…with some degree of independence of the design of others..." (Simon 2019).

In the BMS case, this includes designing modular data structures, configuration management pipelines, and digital workflows that enact the intended enterprise behavior. As these elements cohere, a form of augmented intelligence begins to emerge, not from any one tool, but from the relational alignment of human, procedural, and digital elements acting in concert.

The boundary between architecting and designing is not absolute. It depends on where one stands in the holarchy:

- The enterprise architect may treat governance structures, digital workflows, data platforms, and stakeholder engagement as primary design elements.
- The digital engineering team may focus on model-based processes, digital threads, and lifecycle frameworks as design challenges.
- At the subsystem level, engineers or data modelers may shape interfaces, schemas, and logic that realize these capabilities.

Each actor engages with a different level of scale and responsibility, but all operate through the lens of FFF. What distinguishes their roles is orientation:

- *Architecting* focuses on how the system engages and aligns with its external context.
- *Designing* focuses on how the internal elements realize the system and preserve its integrity.

Elegance arises when this holarchic relationship is respected. When architecting provides a clear framework of the whole, and designing responds with a sufficient and purposeful specification for the components, the result is a system that:

- Exhibits clarity without rigidity,
- Supports emergence without chaos, and
- Delivers complexity without complication.

Such coherence must be cultivated through reciprocal awareness, where architecting anticipates the needs of design, and designing remains attentive to the broader purpose.

WHEN COHERENCE FAILS: DIAGNOSING PATHOLOGICAL EMERGENCE

Emergence is often celebrated as a hallmark of innovation, adaptability, and systemic capability. Yet not all emergent behavior is beneficial. Just as elegance arises from coherent alignment across FFF, *pathological emergence* occurs when complexity produces outcomes that are structured, but no longer coherent with intent, purpose, or context (Mobus and Kalton 2014).

Bar-Yam et al. (1998) observes that in high-dimensional systems, small changes in state or initial conditions can lead to disproportionately large and often unpredictable behavioral shifts. When system transparency is low, or feedback mechanisms are delayed, suppressed, or distorted, control becomes difficult, and alignment between internal configuration and external relevance begins to fray. These dynamics do not necessarily indicate system breakdown; rather, they often reflect systems that are coherently generating unintended or misaligned results.

Importantly, pathological emergence does not always stem from internal design failure. As Troncale (2013) emphasized in his comprehensive treatment of systems pathologies, such failures often arise when the same universal processes that enable systemic health, feedback, cycling, regulation, integration, become impaired or distorted. He classified these dysfunctions into recurring "system diseases". These pathologies are patterned, not random. They occur even when a system remains internally consistent but is no longer attuned to the evolving context or its intended role. They may emerge:

- When the environment shifts in unforeseen ways,
- When stakeholders reinterpret the system's interfaces or outcomes,
- When nature exerts forces, the system was not designed to absorb,
- Or when services and products are extended, misused, or reframed, resulting in unanticipated risks or harm.

These are not simply errors; they are emergent distortions of systemic logic. The challenge is not disorder, but misplaced order. As a practical and classic example, organizational silos. Within a BMS, divisions or units optimize locally but fail to integrate effectively across the enterprise. Each silo maintains internal coherence, but the absence of shared interfaces, feedback loops, or aligned goals leads to systemic fragmentation. The result is duplicated effort, delayed decision-making, loss of situational awareness, and a failure to realize enterprise-level intelligence. This pathology exemplifies how coherence at the part level can coexist with incoherence at the whole, undermining efficiency, effectiveness, agility, and resilience of the overall enterprise.

The following, common patterns of pathological emergence that follow are applied expressions of systemic pathology, informed by Troncale (2013). While Troncale identified recurrent dysfunctions in feedback, cycling, regulation, and integration (e.g., cyclopathologies and cyberpathologies), the list below translates these concepts into forms commonly encountered by the author in organizational, technological, and socio-technical systems:

- Overfitting: Fit becomes too narrowly tuned to past or static conditions, reducing adaptability.
- Gold plating: Form accumulates unnecessary complexity, obscuring purpose, and increasing fragility.
- Runaway feedback: Signals amplify recursively, driving volatility or saturation, as seen in financial crashes or information cascades.
- Functional distortion: A system continues to operate, but its outputs now conflict with its original intent, manifest in bureaucracies, misaligned AI, or misused innovations.
- Coherence collapse: The system's internal structure fragments under the weight of unresolved contradictions or scaling mismatches.
- Unintended externalization: The system performs as designed, but its impact on the environment produces harm, such as ecological degradation or social polarization.

Elegance does not imply omniscient control. Rather, it embodies clarity of intent, sufficiency of structure, and continuous responsiveness to change. The FFF triad enables system architects and engineers to diagnose and realign:

- Is the system's Form still enabling its intended Function?
- Has the Function drifted from the required Fit with the environment?
- Are new interactions producing effects outside the original frame of meaning or responsibility?

In this light, *elegance is not just a design aspiration, it is a systemic capacity for realignment*. It includes the awareness and capability to reframe assumptions, reassess system boundaries, and revisit purpose. Systems pathology, as Troncale envisioned, is not a marginal concern, it is foundational to ensuring adaptive resilience in a complex world.

CONCLUSION: CULTIVATING ELEGANCE IN SYSTEMS PRACTICE

In a world of growing interdependence and rapid change, systems engineering must move beyond controlling complexity to cultivating coherence. This requires more than technical mastery, it demands a deeper systems literacy: the ability to see patterns, recognize misalignments, and act at the level of relational structure (Mobus and Kalton 2014, Simon 2019). Systemic elegance is not a stylistic preference or aesthetic flourish. It is a guiding principle that affirms three key insights:

1. Complexity is not the enemy, it is the medium of emergence and adaptation (Anderson 1972).
2. Emergence is not magic, it is the natural result of systemic interaction and structure (Von Bertalanffy 1968).
3. Elegance is not simplicity, it is contextual *sufficiency*: when Fit, Form, and Function align to support purpose with clarity and adaptability.

The FFF triad thus serves as a systemic compass, a way to diagnose misalignments, assess emergence, and steer toward resilient coherence. As argued earlier, this model expresses the recursive nature of holarchies, where each subsystem is both a part and a whole (Koestler 1968). Whether designing a product, structuring a team, or architecting an enterprise, the question remains the same: Does this system support the right kind of emergence, in the right context, for the right purpose?

When this alignment is achieved, elegance arises, not as control, but as a kind of attunement. Elegant systems feel alive: they adapt, inform, and endure. In an era that demands both innovation and responsibility, systemic elegance offers a path forward, grounded not in complexity for its own sake, but in the wisdom of purposeful design. ■

DISCLAIMER

The author acknowledges the significant intellectual support provided by OpenAI's ChatGPT4.0-o in the development of this paper. The AI model was used extensively to synthesize ideas, refine theoretical constructs, and draft content in close collaboration with the author. All conceptual inputs were reviewed, curated, and integrated by the human author, who retains full responsibility for the content and conclusions presented herein.

REFERENCES

- Anderson, P. W. 1972. "More Is Different." *Science* 177 (4047): 393–396. https://doi.org/10.1126/science.177.4047.393.
- Ashby, W. R. 1961. *An Introduction to Cybernetics*. Chapman & Hall Ltd.
- Bar-Yam, Y., S. R. McKay, and W. Christian. 1998. "Dynamics of Complex Systems" (Studies in Nonlinearity). *Computer in Physics* 12 (4): 335–336. https://doi.org/10.1063/1.4822633.
- Dumas, A., and H. Mintzberg. 1991. "Managing the Form, Function, and Fit of Design." *Design Management Journal (Former Series)*, 2(3), 26–31.
- Handbook, Military 2001. *Configuration Management Guidance*. MIL-HDBK-61A (SE) Department of Defense, US.
- Koestler, A. 1968. *The Ghost in the Machine*. Macmillan.
- Mobus, G. E., and M. C. Kalton. 2014. *Principles of Systems Science*. New York, US-NY: Springer. https://books.google.co.uk/books?id=twBNBQAAQBAJ.
- Sillitto, H., J. Martin, D. Mckinney, R. Griego, D. Dori, D. Krob, P. Godfrey, E. Arnold, and S. Jackson. 2019. *Systems Engineering and System Definitions* INCOSE.
- Simon, H. A. 2019. *The Sciences of the Artificial*, reissue of the third edition with a new introduction by John Laird. MIT Press.
- Troncale, L. 2013. "Systems processes and pathologies: Creating an integrated framework for systems science." INCOSE International Symposium 23 (1): 1330-1353, Philadelphia, US-PA, 24-27 June.
- Von Bertalanffy, L. 1968. *General System Theory: Foundations, Development, Applications*. New York, US-NY: George Braziller.

ABOUT THE AUTHOR

Gary Smith is a senior expert systems engineer at Airbus Defence and Space, where he leads the architecture of engineering processes and the digital transformation of the division. He is an INCOSE certified expert systems engineering professional (ESEP) and a senior editor of the Systems Engineering Body of Knowledge (SEBoK). A long-standing advocate for integrating system science with engineering practice, Gary has served as vice president for systems practice at the International Society for the Systems Sciences (ISSS) and as the society's liaison to INCOSE. He has contributed extensively to the development of coherent frameworks for systemic inquiry, practice, and education, bridging complex domains from aerospace and defence to healthcare and sustainability. Now serving as past president of the ISSS, Gary continues to champion collaboration across disciplines and professional societies. He is a leading contributor to the emergence of a unified system science enterprise, focused on establishing shared foundations that support both scientific understanding and practical application. His work emphasizes the triad of fit, form, and function as a basis for achieving coherence, managing complexity, and cultivating systemic elegance.

The Purposeful Evolution of Systems Engineering Heuristics Using I-SHARE

Javier Calvo-Amodio, javier@uidaho.edu; **Harington Lee,** haringtonl@gmail.com; **Veronika Shteingardt,** veronika-s@campus.technion.ac.il; **Dorothy McKinney,** dorothy.mckinnry@incose.net; and **Dov Dori,** dori@technion.ac.il

ABSTRACT
A heuristic relates to a formulation based on experts' experience, which draws on observed common patterns and serves as a guide in investigating or solving a problem. A transdisciplinary field, systems engineering involves many useful heuristics, as it integrates the gamut of engineering disciplines in defining a system throughout its lifecycle. To be usable, heuristics should be memorable and pithy, and the consequences of applying them should be predictable. To be predictable, a heuristic should provide insights into how and why it works in a particular context. The first step to increase the capability of systems engineering heuristics was the creation of the I-SHARE–INCOSE Systems Heuristics Application Repository, a curated knowledge base of over 600 systems engineering-related heuristics covering systems engineering competencies, lifecycle stages, expertise, operational domains, system attributes, and more. Here, we describe a process for guiding the systems engineering community on how to validate, test, and assess heuristics, and how the systems engineering community can engage with I-SHARE to benefit from the heuristics in it and collectively improve their capabilities.

KEYWORDS: heuristics; heuristic principles; I-SHARE

INTRODUCTION

In response to an INCOSE leadership call in 2019, a group of INCOSE Fellows compiled and curated the INCOSE Systems Heuristics Application Repository (I-SHARE) (Dori et al. 2022). At the same time, the Systems Engineering Principles Action Team produced a set of 15 principles and three hypotheses (Watson et al. 2019, 2022) as an evolution of work produced by the NASA consortium (Watson et al. 2018). When these products were presented at INCOSE events, it became apparent that despite key differences between the intended use and scope of a principle and a heuristic, their definitions did not clarify how they differ. Given the lack of precision in definitions about what constitutes a principle and a heuristic, a "Bridge Team" was established to crystallize the definitions of these two key concepts and help clarify the differences between them. The Bridge Team posits that a principle and a heuristic are special cases of a more general concept— a guiding proposition, which can be one of four types: heuristic, convention, perspective, and model or mechanism. All the guiding propositions have their origins in observed patterns that we discover as we reflect on our practice, community, values, and nature. Any guiding proposition has the potential to increase its scope of application and authority as it is tested in new contexts or as evidence of how and why it works is collected. The capability of a guiding proposition to provide guidance increases as its authority and scope of application increase. A guiding proposition becomes a principle once we gain insight into why and how it works. For a guiding proposition to be considered a principle, it should be science-informed or science-explained. Specifically, a heuristic becomes a heuristic principle, a convention becomes a social principle, a perspective becomes a philosophical principle, and a mechanism or a model becomes a scientific principle. The first three are science-informed principles, while the last one is a science-explained principle. As our focus here is on heuristics, this insight is crucial in disambiguating the meanings of what a heuristic and a principle are, and most importantly, explains why there was significant overlap between the definitions for both. The general process we propose can be applicable to other guiding proposition types.

The research question we focus on here is the following: How can we leverage the I-SHARE repository to enable the systems engineering community to collectively elevate our heuristics in scope, authority, and capability?

To begin answering this question, we link the works by Rousseau et al. (2024) and Dori et al. (2022), by proposing a detailed model of the evolution process of a heuristic into a principle.

Motivation

As the complexity of the solutions we design continues to rise, we need to further develop our methods to resolve the challenges that come with this increase. We begin by adopting the definition of complexity as proposed by Rousseau & Billingham (2025), which is the "internal variety in terms of parts, inter-part relationships, potential states and state-changes (behaviours)" that a system exhibits. We follow by defining systems and systems engineering. A recent overarching definition of a system is "*an arrangement of parts or elements that together exhibit behavior or meaning that the individual constituents do not*" (Dori et al. 2020). This is a broad definition in that it describes both natural and human-made systems, as well as both physical and informatical, logical, or intangible ones. According to INCOSE (2025), an engineered system is "*designed or adapted to interact with an anticipated operational environment to achieve one or more intended purposes while complying with applicable constraints.*"

Based on systems principles and concepts, systems engineering is a *transdisciplinary and integrative approach to enable the successful realization, use, and retirement of engineered systems, using systems principles and concepts and scientific, technological, and management methods.* (Sillitto et al. 2018). The Bridge Team has enriched this definition by incorporating ideas from Frosch (1969) and Griffin (2010), specifying that the purpose of systems engineering is to attain increasingly elegant solutions that resolve increasingly complex challenges. Rather than focusing solely on processes and techniques, attention is paid here to elegant design as value at a higher level of abstraction in the practice of systems engineering. Adopting this perspective can guide systems engineers in using heuristics when addressing complex problems (Pennotti et al. 2024). This systems engineering process begins by establishing the holistic purpose of systems engineering, influenced by stakeholders, their values, success criteria, as well as societal and personal values reflected in their guiding propositions. The process defines actual or anticipated customer needs and required functionality early in the system development cycle. Systems engineering establishes a systematic approach using system-level guiding propositions that determine what needs to be done. Finally, systems engineering's systematic practice is explored by developing an appropriate lifecycle model approach that considers factors including complexity, uncertainty, and change. Requirements are documented and modeled for each phase of the endeavor, then the design is synthesized, and the system is tested and validated. The complete problem and all the necessary enabling systems and services are considered throughout the systems engineering process, maintaining and often increasing its value.

Systems engineering provides guidance and leadership to integrate all the disciplines and specialty groups into a team effort, forming an appropriately structured development process that proceeds from concept to production, operation, evolution, and eventual engineered recycling or disposal. Systems engineering considers the various stakeholders and accounts for both the business and the technical needs of the system users and customers, aiming to provide a quality solution that meets the needs of users and other stakeholders and is fit for the intended purpose in real-world operation. This description, officially adopted by INCOSE, serves as a baseline to show that systems engineering can significantly benefit from the judicious application of appropriate heuristics. The line of thought leading to the initiative described herein is that since systems engineering is a transdisciplinary field of engineering (Sillito et al. 2018), drawing from a variety of sources and arguments, it is likely to benefit significantly from an orderly compilation and curation of heuristics relevant to the field.

HEURISTICS: DEFINITION AND EMPLOYMENT

The word *heuriskein* in ancient Greek and *heurisricus* in Latin mean "to find out, discover," providing the gist of heuristic as something that is found through discovery rather than logic or thought. *Heuretics* is the branch of logic that treats the art of discovery or invention; that which treats the conditions of knowledge that lie in nature, not of the thought itself. Romanycia and Pelleti (1985) reviewed the history of the heuristic concept in artificial intelligence (AI) from the perspective of four dimensions: uncertainty of outcome, basis in incomplete knowledge, improvement of performance, and guidance of decision-making. A recent faithful account on heuristics (Cherry 2022) describes them as "*mental shortcuts that allow people to solve problems and make judgments quickly and efficiently. These rule-of-thumb strategies shorten decision-making time and allow people to function without constantly stopping to think about their next course of action.*"

A general definition of a heuristic is "*a mental shortcut that allows an individual to make a decision, pass judgment, or solve a problem quickly and with minimal mental effort*" (Psychology Today 2025). Heuristics can potentially reduce the burden of decision-making and free up limited cognitive resources. While potentially highly beneficial, heuristics can also be costly, as they can lead users to miss critical information or cause unjust or biased actions. In the context of mathematics, Polya (1945) defined the aim of heuristics as the study of methods and rules of discovery and invention. He noted that Descartes and Leibnitz attempted to build up a system of heuristics and viewed heuristics as fallible statements that contrast with deductive reasoning. *Heuristic reasoning* is reasoning not regarded as final and strict but as provisional and plausible, aimed at discovering a solution to a problem. Before obtaining certainty, we often need the provisional, so a temporary solution can be satisfied with a plausible guess.

In the context of psychology, heuristics are considered "*rules-of-thumb that can be applied to guide decision-making based on a more limited subset of the available information.*" As heuristics are based on a subset of the information needed for a fully reliable decision, it is assumed that they facilitate more timely decision-making than strategies that require more information.

Heuristics are important in the emergence of new scientific and engineering fields. They capture the successful approaches and those to avoid in the engineering of a system. However, heuristics are context-sensitive and must be applied with judgment to the current and new system contexts (Maier and Rechtin 2009). This context sensitivity is a powerful aspect of heuristics, allowing them to be developed organically from specific system examples and then applied to future systems with a similar context. For example, in the context of the financial world, where financial professionals use a heuristic approach to speed up analysis and investment decisions, heuristics are described as "*mental shortcuts for solving problems in a quick way that delivers a result that is sufficient enough to be useful given time constraints.*"

Maier and Rechtin (2009), whose book focuses on architecting, provided the following definition of heuristics: "*A Heuristic is a guideline for the conduct of architecting; lessons learned expressed as a guideline; a natural language abstraction of experience.*" They add: "*Heuristics … [are] abstractions of experience, … trusted, nonanalytic guidelines for treating inherently unbounded, ill-structured problems. They are used as aids to decision-*

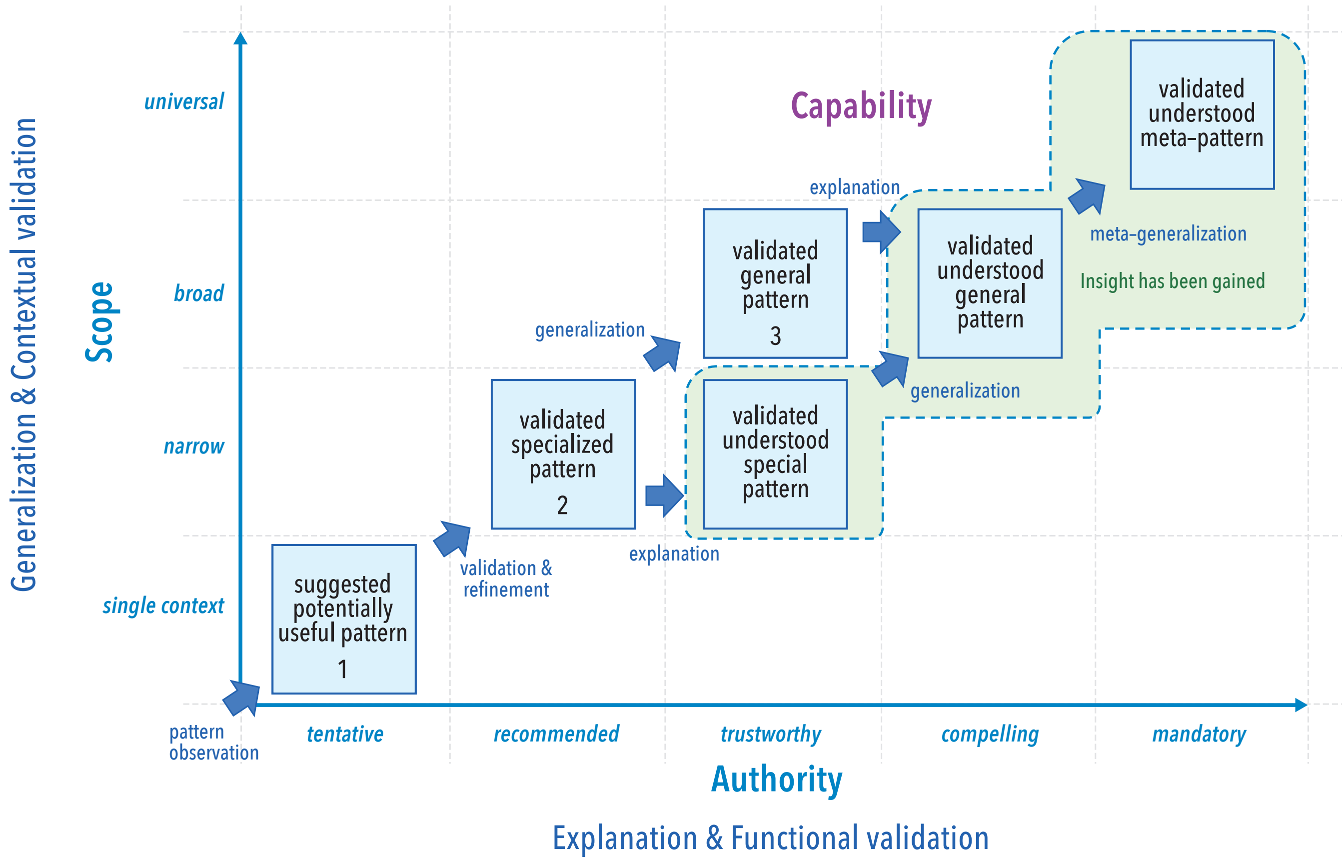


Figure 1. The evolution of systems engineering guiding propositions (adapted from Rousseau et al. 2024)

making, value judgments, and assessments" (p. 31). Examining the definition of this heuristic with the general one, as well as those of finance and psychology cited above, we see the commonality: all emphasize the idea that heuristics aim to help solve problems quickly and effectively based on the largely informal experience of professionals who have seen and done a lot in their field. Their experience puts them in a position to generalize, abstract, and provide "words of wisdom" to less experienced professionals. For systems engineering, this is perhaps even more true than for any other profession because systems engineering practitioners combine engineering and art, as implied by the book title *The Art of Systems Architecting* (Maier and Rechtin 2009).

GUIDING PROPOSITIONS

In the context of systems engineering, we can think of heuristics as a specialized form of a more general kind of guidance or guiding propositions. The Bridge Team has proposed a framework, presented in Figure 1, for understanding the origin and evolution of guiding propositions, including models/mechanisms, principles, heuristics, conventions, and perspectives. Generally, a guiding proposition is "*a rule that can guide our purposeful judgment-making or action-taking within a context.*" A guiding proposition, or guidance, is an explicit representation of a useful pattern distilled from our experiences. A guidance can evolve according to the range of contexts, or scopes of application, in which it applies and how persuasive or authoritative it is in its application. While scope and authority are somewhat independent, capability is the combination of both. As the scope of application of a guidance widens and its authority increases, the guidance becomes more capable of predicting the consequences of its application (Rousseau et al. 2024, p. 27).

THE EVOLUTION OF SYSTEMS ENGINEERING GUIDING PROPOSITIONS

Engineering was born out of the need to resolve or mitigate a challenge or concern, or to satisfy an opportunity or interest. From the need to provide reliable shelter to the ability to fly, engineers have observed, tested, and honed methods to improve the design and implementation of solutions. Brainpower alone is insufficient to resolve the challenges humanity faces to reliably thrive as a species. Humans have survived for so long because they established societies around traditions and cultures that are fit for the environment in which they live. Traditions and cultures employ heuristics, conventions, perspectives, and mechanisms that have been developed over time, providing invaluable guidance for making judgments and taking actions in known environments. Thus, guiding propositions inform our methods to resolve challenges, concerns, opportunities, and interests. Once we gain insight into how or why a guidance works, we can make impressive progress. Consider, for instance, the evolution of guidances related to the use of fire as a heat source to alter our environment. Eventually, we identified patterns that guided how to use fire for cooking, creating art and weapons, culminating in scientifically derived principles that have enabled such modern manufacturing methods as arc welding, laser cutting, and advanced electronics.

Why evolving systems engineering heuristics into principles matters?

Systems engineering needs to understand the evolution of guiding propositions of all kinds. We have elected to focus first on heuristics for pragmatic reasons: We already have a large set of heuristics curated in I-SHARE. Developing the process through which we can evolve our heuristics

Table 1. String diagram notations definitions

Notation	Definition
Tensor ⊗ or "circled times"	Example: A ⊗ B represents "an A and a B".
Exclusive Or ⊕ or "circled plus"	Example: A ⊕ B represents "either an A or a B"
Split Diamond ◇<	A split diamond corresponds to a map A ⟶ A⊕, and this should be thought of as a (Boolean) test on A. In natural language, "If t(a) is a Boolean test on a, a:A, then either t(a) is true or false"
Merge Diamond ◇>	A merge diamond is a map A ⊕ B ⟶ C, corresponding to the following natural language: "If I have (either A or B), then I have C."

Table 2. Explanations for the notations used in Figure 2

Notation	Explanation
Heuristic = {Scope, Authority}	A heuristic has (the attributes) Scope and Authority.
Scope = {context1, $context_2$,...,$context_n$}	Scope is a set of n (or has n values of) contexts.
Authority = {scientific insight, socio-cultural agreement}	Authority has (consists of) a level of scientific insight and a socio-cultural agreement.
Problem = {context, socio-cultural agreement}	A problem is (comprised of) a context and a socio-cultural agreement.
SE ⊗ Context ⟶ Experience Level	For every Context, a SE (systems engineer) has a certain value of Experience Level.

can potentially help establish a broader, more generalized process applicable to all kinds of guiding propositions.

A benefit of discovering the process of evolving heuristics is that as new challenges are encountered, heuristics can be useful also in unknown environments. However, applying a heuristic in making judgments or acting in a different context can be more reliable if it is elevated to a heuristic principle, which can be attained if insights into why and how the heuristic works have been gained. Heuristics can thus serve as enhancers of problem-solving skills, distilling experience into useful activity patterns that help resolve challenges. However, if we do not possess insights into how heuristics work, we are likely to abandon practicing that heuristic, as we will have nothing to revise and therefore nothing to learn. In his Theory of Profound Knowledge, Deming (2018) states that a theory contains knowledge if it can predict a future outcome, with a risk of being wrong, and that it fits without failure observations of the past. A guiding proposition is equivalent to a theory in Deming's terms: It should lead to the prediction of a future outcome, enable learning, and therefore raise new questions. This is the challenge that, as systems engineers, we face: We tend to work in one-off complex systems with different contexts or environments, for which a past heuristic may not apply as is, calling for a revision or adaptation to the new context. If we have insights into how and why a heuristic works in the context in which it was discovered and developed, we can assess and, if possible, adapt the heuristic to the new context in a rigorous manner.

A REFERENCE STRUCTURE TO GUIDE THE PROCESS OF EVOLVING HEURISTICS

Systems engineers use various methods for creating models. In this work, we have selected a category theory modeling tool called string diagrams, which provides a formal language for representing processes and relationships at a spectrum of abstraction levels from high to low (Breiner et al. 2023).

The intent of our model is to provide a general reference structure that describes at a high level of abstraction the process depicted by the arrows (lines) in Figure 1. The decision to only represent a high level of abstraction by depicting a schema is pragmatic and inspirational. It is pragmatic because, at this point, it is not feasible for us to provide semantic details about all the possible use cases or instantiations. It is inspirational because it serves as a call to action for all systems engineers to discover and develop cases for such applications, helping us refine the proposed schema while evolving systems engineering heuristics. Table 1 provides the notations used by string diagrams in this work.

Table 2 contains explanations for the notation used in Figure 2. Figure 2 is a string diagram model that represents the details abstracted by each one of the blue arrows in Figure 1.

Following these arrows leads to different evolution paths a heuristic can take depending on its maturity level. The String Diagram starts on the left-hand side of Figure 2 with a problem identified by a systems engineer (SE ⊗ problem). The systems engineer then consults the I-SHARE repository to identify a candidate heuristic to help solve the problem (heuristic ⊗ problem ⊗ SE). After applying the heuristic, the systems engineer will either solve or fail to solve the problem by applying the heuristic (SE ⊗ (problem ⊕ solution) ⊗ heuristic; in natural language, this expression can be read as "an SE and a heuristic and either a problem or a solution"). If at this point the systems engineer finds a solution, the pair of the solution and the heuristic (solution ⊗ heuristic) that helped solve the problem is recorded, and the systems engineer terminates her role — note that the (solution ⊗ heuristic) will be entered into I-SHARE as noted in the left side of the string diagram. If the heuristic fails to deliver a solution, the systems engineer can revise the heuristic and try again. Conversely, or after one or more attempts to revise the heuristic, it may be decided that the heuristic does not work in the context, or it does not satisfy the expectations (contained in the social-cultural agreement), or is not trustworthy enough.

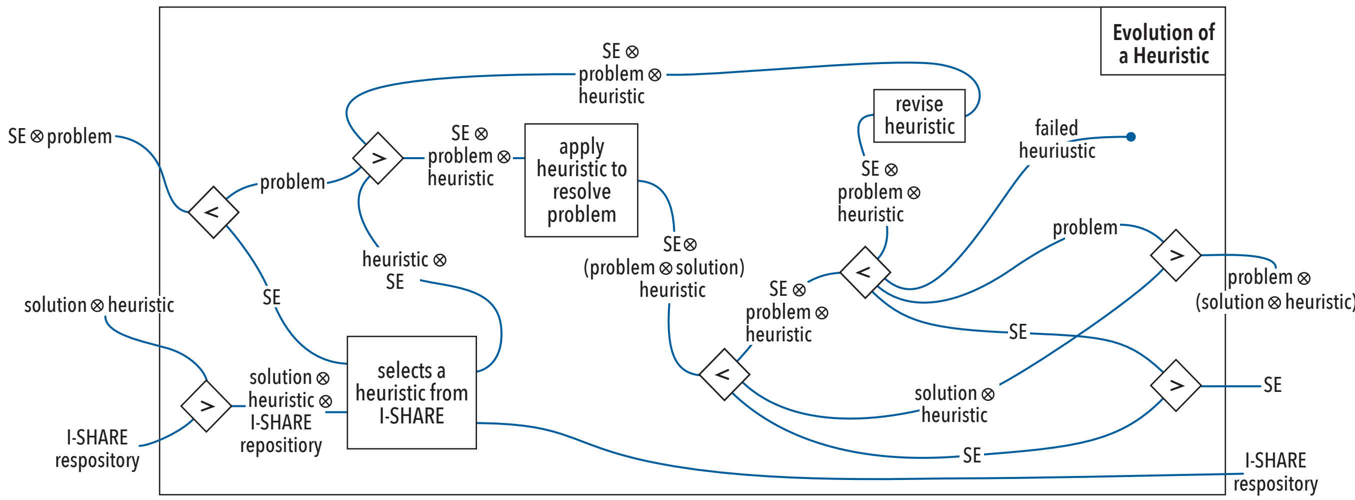


Figure 2. A string diagram depicting the evolution of a heuristic using I-SHARE

Note that the attributes identified in Table 2 for each object (heuristic, SE, solution, I-SHARE repository, problem) may be updated after each function (Selects a heuristic from I-SHARE, apply heuristic to resolve problem, and revise heuristic) in Figure 2.

The forging of Japanese swords is an example of the evolution of a heuristic, demonstrating that reflections on traditional practices can be found in the forging of these swords. Famous for their endurance due to a combination of hardness and flexibility, Katanas were crafted by following sophisticated heuristics, developed over centuries of experience. The heuristics would guide craftsmen in selecting the right materials and assessing the steel quality (resolving problems) through its sound and fracture patterns when struck, as well as its color. Similarly, folding techniques, hardening, quenching, finishing, and testing were achieved by following constantly refined heuristics. These time-honed heuristics used by Japanese craftsmen are now science-informed as we possess scientific knowledge that provides insights into how and why the heuristics work.

The set of problems experienced by Japanese craftsmen when forging katanas was specific to the context of forging katanas (See Figure 3). This allowed them to work on increasing the authority of their heuristics, thus gaining insights into how or why their heuristics worked. However, the knowledge gained would also be shared in different metal forging contexts, thus gaining insights into their applicability in other contexts. The process also enabled Japanese craftsmen to discard failed heuristics and refine them through experimentation, reformulation, etc.

Our call to the systems engineering community is to engage with I-SHARE to select existing heuristics, test, validate them, report back the insights gained, and propose new or modified ones.

I-SHARE STRUCTURE AND REVIEW PROCESS

I-SHARE Development Methodology

I-SHARE is a curated set of systems engineering heuristics that follows the steps of the systems engineering process and delivering a system that aims to meet or exceed its requirements and expectations. It aims to assist the decision-making process for and delivering a system that meets or exceeds its requirements and expectations. In this section, we provide an overview of the I-SHARE development methodology.

The INCOSE heuristics activity was launched in 2020, as described by Dori et al. (2022) to devise *I-SHARE – INCOSE Systems Engineering Heuristics Repository* to be shared among practicing systems engineers. The requirements for a heuristic to be included in I-SHARE were the following: (1) change an action that might otherwise occur without the heuristic, (2) be pithy and memorable so it can be recalled when the challenge arises, (3) express an abstract phenomenon in a

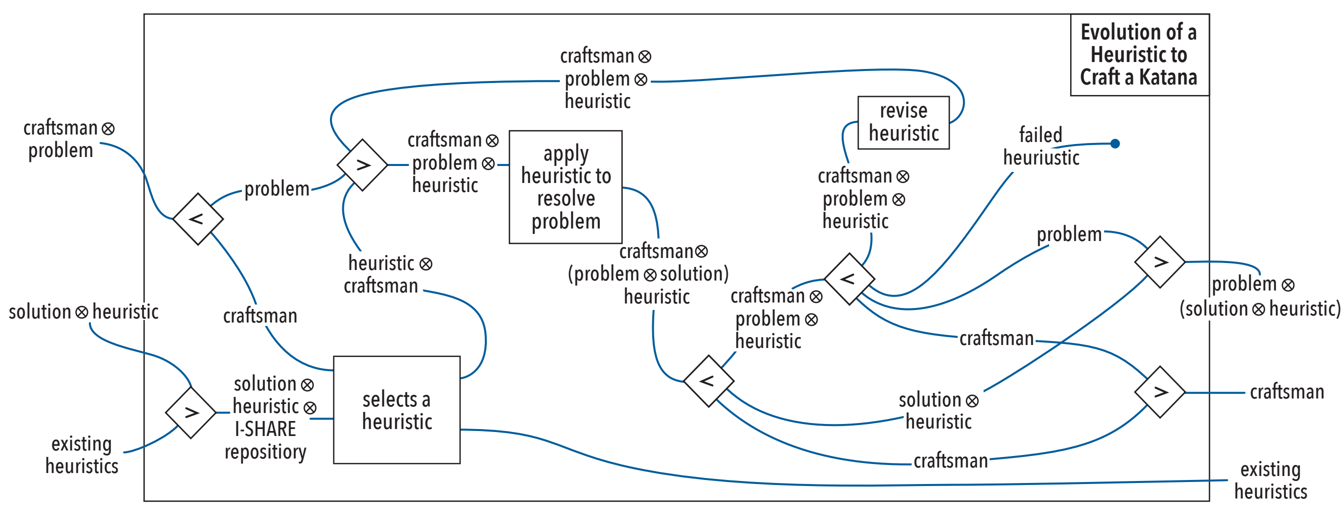


Figure 3. General. evolution process of heuristics to craft katanas

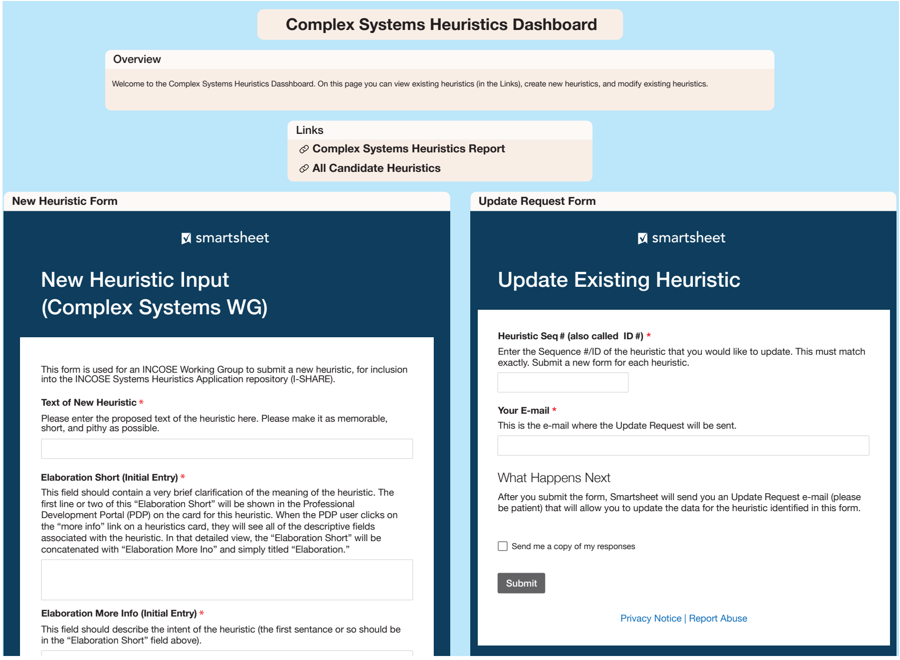


Figure 4. A heuristics dashboard example for the Complex Systems WG

simple, understandable way, (4) emphasize usefulness over precision or universality, (5) be curated to remain relevant and referenced by systems engineers and others, and (6) complement and support the challenges of the *Systems Engineering Principles*, under development by the Future of Systems Engineering (FuSE) program to realize the *Systems Engineering Vision 2035* (INCOSE2021).

Users and Stakeholders.

The primary users for the systems engineering heuristics knowledge base are the following groups: (1) systems engineers, in performing systems engineering tasks throughout the system lifecycle, (2) systems thinkers, in developing systems engineering methods and tackling general systems-related problems, (3) managers and decision-makers, in evaluating design proposals and performing decision making, and for improved oversight of systems engineers' work, and (4) systems engineering training and development programs to augment organizational training materials with I-SHARE to improve the quality, efficiency and competency of their systems engineering teams. Secondary users and stakeholders are the wider community, who may find the heuristics informative or useful, including (1) engineers in other professions, with emphasis on software engineers, (2) the academic community, to enrich systems engineering curricula and course materials, (3) INCOSE, to enhance its reputation and support its mission, and (4) other international organizations, such as IEEE and ISO, to consider using I-SHARE heuristics in standards and guidelines.

Objectives and use cases.

The heuristics included in I-SHARE are expected to (1) find a solution to or identify the important factors to focus on in addressing a complex or difficult problem, especially a "wicked" one, or identify the important factors to focus on in addressing a complex, difficult, or "wicked" problem, and (2) reduce the time needed to make a good decision or choice, or quickly find a good solution by drawing on best practices. Several use cases and scenarios that illustrate typical I-SHARE usage are provided by Dori et al. (2022), including (1) supplementing expert review and feedback, (2) providing guidance from lessons learned in systems engineering, (3) synthesizing and evaluating a systems engineering product, (4) searching relevant heuristics, and (5) improving organizational culture.

Selection and Review Process

I-SHARE uses the work management tool SmartSheet as its database management service (See Figure 4). In SmartSheet we have created dashboards for each participating INCOSE working group (WG) to simplify the process. Each dashboard contains two forms: new heuristics form, and update request form. On the new heuristics form, WG members can propose new candidate heuristics that are added to the database for processing. On the update request form, they can propose changes to candidate or published heuristics. (Note that SmartSheet will be replaced with another tool in the near future as INCOSE's license will expire in May 2026.)

Figure 5 shows the I-SHARE heuristic lifecycle, from selection through review and refinement to publication shows the section, review, and refinement process. Table 3 presents the heuristics selection criteria.

SHARE Heuristics Attributes

As the I-SHARE team concluded that collecting heuristics alone was not sufficient, they decided to process them for making them more useful. This processing consists of rewording the heuristic as necessary to make it short, pithy, and memorable, and adding for each heuristic in I-SHARE the following attributes:

- **Elaboration** – Since heuristics are supposed to be pithy and memorable, they do not provide a complete descrip-

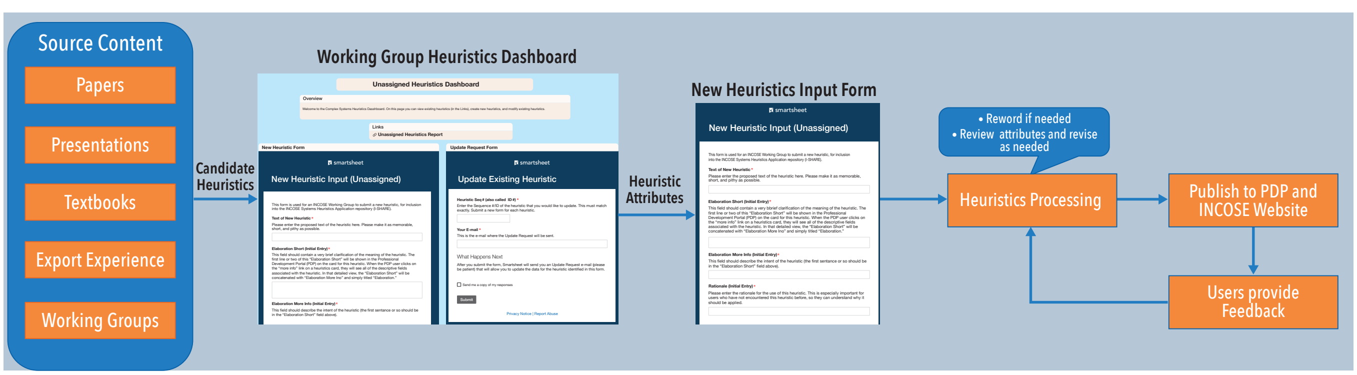


Figure 5. INCOSE heuristic selection, review, refinement, and publication process

Table 3. The heuristics selection criteria

No.	Criterion	Rationale	Comments
1	The heuristic must be focused, clear, and concise.	Even if the original language is not clear and succinct, we should be able to rephrase it in clear and succinct terms. To this end, we allow heuristics rephrasing.	The heuristic should be narrower and more focused than common sense or "motherhood and apple pie." They should abstract experience, so lessons learned are passed to less experienced practitioners.
2	The heuristic's source must be authoritative.	Ideally, each heuristic should be published or suggested by a recognized expert in the systems engineering field and supported by at least two others.	Newly expressed heuristics will probably need to be vetted more thoroughly than ones that have been published and are recommended by one of the I- SHARE team members.
3	The heuristic's use can be clearly articulated.	Unless we can state when or how a heuristic can or should be used, it is not likely to serve our intended purpose in promulgating I-SHARE.	Uses could be for a specific of the systems engineering life-cycle phase or a specific type of system, such as complex, socio-technical, cyber-physical, or system-of-systems.
4	The heuristic must make sense in its original domain or context	The heuristic should also apply more widely, using heuristics extrapolation (Maier and Rechtin 2009).	See the explanation on heuristics extrapolation by Maier and Rechtin (2009).
5	The heuristic should be capable of being applied beyond its original context.	The heuristic should be useful in solving or explaining more than the original problem from which it arose.	The heuristic should be generalizable so it becomes applicable across multiple domains.
6	The heuristic can be easily rationalized in a few minutes of talk or in one medium-length paragraph.	Terseness eases subsequent publication and use.	Supporting material may need to be developed before the rationalization can be judged.

tion and explanation of the guidance they offer, so additional elaboration is needed to enable systems engineers to understand the meaning and intent of the heuristic. There are two kinds of elaborations: short and long.

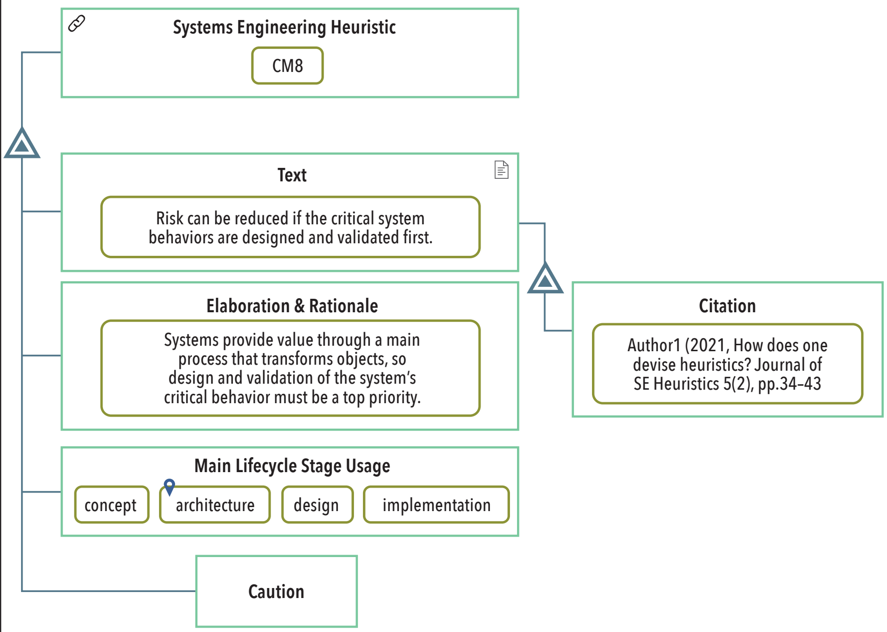
- **Rationale** –The rationale underlying a heuristic is provided to help systems engineers understand the gist of the heuristic, especially targeting users who have not encountered this heuristic before.
- **When to use** – Many heuristics apply only under certain circumstances or in specific stages of an engineering effort. This information aims to help users understand whether this heuristic applies to their specific circumstances.
- **Specific Processes** – These are suggested ISO/IEC/IEEE-15288 technical processes and technical management processes where the heuristic is best applied.
- **Cautions** – The application of a heuristic is not "one size fits all," so this information is intended to assist users in applying the heuristic in ways that are most likely to reduce risks and increase the probability of success in the systems engineering effort.
- **Why do I care?** – This information is intended to tell users what concerns or risks the heuristic addresses, helping them understand the heuristic importance and significance.
- **Expertise needed** – Some heuristics require special expertise to be applied successfully. If special expertise is needed, the special expertise is explained, and possibly how to obtain the needed expertise.
- **Citation** – Many of the writings from which heuristics were drawn discuss in detail the basis for the heuristic, and in many cases examples of its application are provided. If available, published materials are cited, including internet URLs, enabling users to find additional information on the foundations and origins for the heuristic and guidance for using it.
- **Source** – Many of the heuristics were contributed by individuals who, while spending years applying specific heuristics in practice, may not know all the published sources about them. Since many of the contributors are active INCOSE members, users may be able to contact them to learn more about the application of the heuristic.

I-SHARE Usage

In this section, we discuss the usage of heuristics with an object process methodology (OPM) model example, as well as how to access the heuristics database.

An OPM Model Example

Let's use one example systems engineering heuristic and walk through the definition, elaboration, and a use case. The chosen heuristic is: "*An interface can be accounted for only when it is owned.*" To provide the elaboration and rationale, we note that interface issues are the most frequent cause of problems in system integration. Therefore, interface management is a high return on investment (ROI) activity, and it is critically important to assign a team or individual as the owner and the one who is accountable for every interface within the system or product.



- *Systems Engineering Heuristic is CM8.*
- *Text of Systems Engineering Heuristic is "Risk can be reduced if the critical system behaviors are designed andd validated first."*
- *Elaboration & Rationale of Systems Engineering Heuristic is "System provide value through a main process that transforms objects, so design and validation of the system's critical behavior must be a top priority."*
- *Main Lifecycle Stage Usage of Systems Engineering Heuristic can be architecture, concept, design, or implementation.*
- *Main Lifecycle Stage Usage of Systems Engineering Heuristic is currently at state architecture.*
- *Citation of Text is "Author1 (2021), How does one devise heuristics? Journal of SE Heuristics 5(2), pp. 34–43."*
- *Systems Engineering Heuristic exhibits Caution, Elaboration & Rationale, Main Lifecycle Stage Usage and Text.*
- *Text exhibits Citation*

Figure 6. An OPM model of a heuristic example

OPM (Dori 2002, 2016) can be useful to specify heuristics both visually and textually. To demonstrate this, we create an OPM model example of a particular heuristic whose text is "*Risk can be reduced if the critical system behaviors are designed and validated first.*"

Figure 6 is the OPM model of the example heuristic, showing the object-process diagram (OPD) at the top and the corresponding, automatically generated text in a subset of English—object process language (OPL) at the bottom. The object at the top of the OPD is the heuristic with its identifier (CM8). This model fact is expressed also by the OPL sentence *Systems Engineering Heuristic* is *CM8*. For demonstration sake, connected to this object with an exhibition-characterization link (the black-in-white triangle) are some of the attributes listed in the previous section with their values that are fit for the chosen heuristic instance, for example, *Elaboration & Rationale of Systems Engineering Heuristic is "Systems provide value through a main process that transforms objects, so design and validation of the system's critical behavior must be a top priority."*

This model is a small demonstration of the potential that can be gleaned from automating the modeling of the heuristics so they can be queried and related to each other in a large, interconnected model. This is a topic of future research and development.

Accessing the Heuristics Database

The heuristics database is currently viewable on INCOSE's Heuristics webpage:

- First log in to www.incose.org (because heuristics are only available to INCOSE members)
- Then open a new browser window or tab and go to www.incose.org/heuristics

The upper part of this web page has a description of heuristics and how INCOSE has chosen the ones to present, followed by a description of supporting information for each heuristic. The web page then lists several possible use cases for the heuristics with descriptions of each use case available with a pull down. Near the bottom of the page, you will find a viewable database of all the published INCOSE heuristics. INCOSE is currently revamping its IT infrastructure, which may impact how these heuristics will be displayed in the future.

If you would like to download the heuristics to examine it offline, click on the three dots next to "(View Only)" at the top left side of the SmartSheet viewing window, as shown in Figure 7.

This will bring up a drop-down menu, from which you should select Export, and then pick one of the four file formats to export to, as shown in Figure 8. We recommend Excel and PDF formats as the most likely to be usable.

HOW TO GET INVOLVED

The heuristics team collaborates directly with INCOSE working groups to gather candidate heuristics. Here are some of the working groups that we have worked with or reached out to:

- Complex Systems
- Resilience
- Systems Security Engineering
- Agility
- Adaptability
- Risk Management
- Measurement
- Human Systems Integration (HSI)

If you are part of an INCOSE working group and have candidate heuristics that you would like added to I-SHARE, contact the authors. For new working groups, the team will create a SmartSheet dashboard and host an onboarding session. Additionally, if you want to test in new contexts the current heuristics contained in I-SHARE, contact the authors.

CONCLUSIONS

A change in business as usual is needed within the systems engineering community if we are to advance the capability of our heuristics. The change requires a coordinated and formal approach to the application and revision of the scope of application and authority of heuristics. The authors call on INCOSE and the systems engineering community at large to leverage I-SHARE as the platform to enable a collaborative and coordinated effort to evolve our heuristics. ■

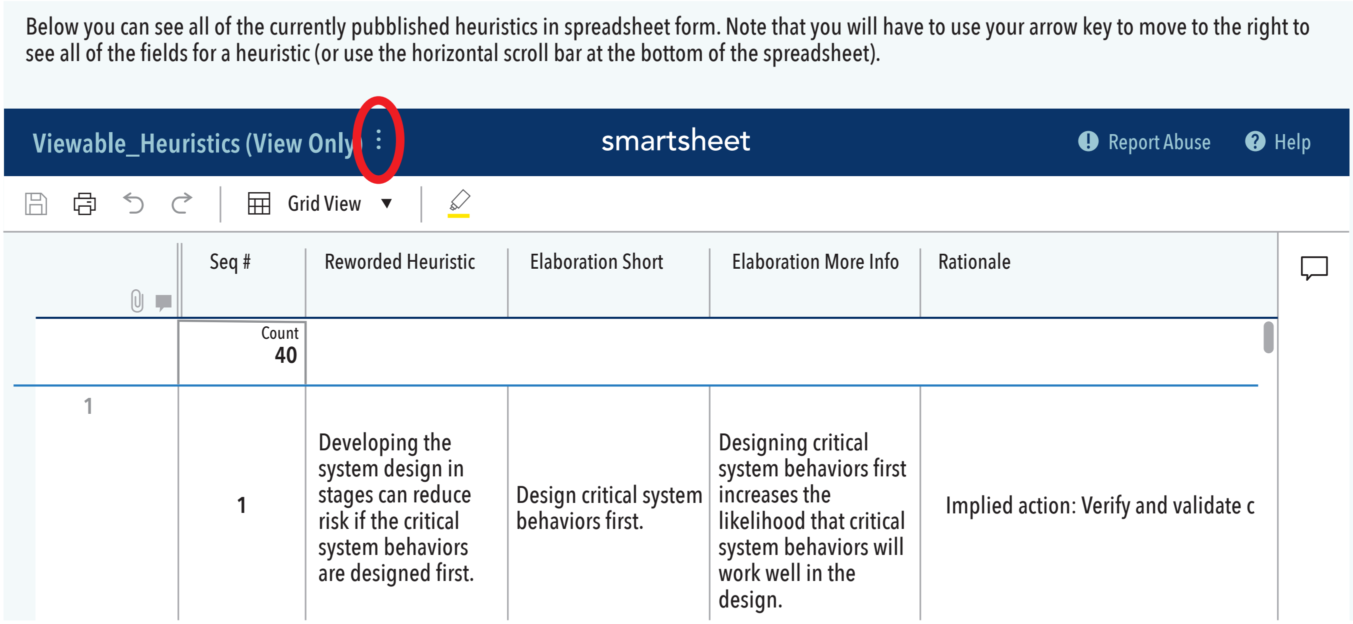


Figure 7. *Click on the three dots next to (View Only) at the top of the window to bring up the drop-down menu*

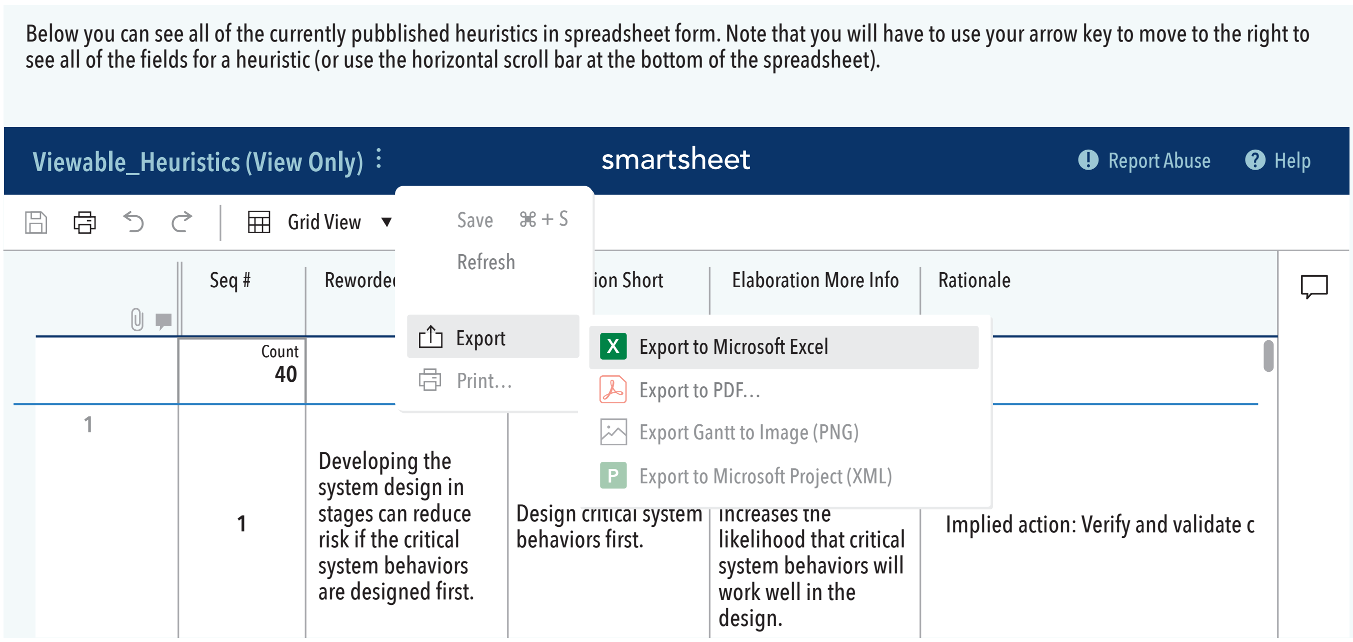


Figure 8. *Choose one of the four file formats to export the selected heuristics.*

REFERENCES

- Breiner, S., E. Subrahmanian, and R. Sriram. 2023. "Category Theory." In A. Madni, N. Augustine, and M. Sievers, *Handbook of Model-Based Systems Engineering* (pp. 1-44). Cham: Springer.
- Cherry, K. 2022. What Are Heuristics? Very Well Mind, Nov. 8.
- Deming, W. E. 2018. *The new economics for industry, government, and education*. MIT Press.
- Dori, D., D. McKinney, G. Wang, and S. Jackson. 2022. I-SHARE – INCOSE Systems Heuristics Application Repository: Sharing Systems Engineering Knowhow and Experience. 33rd INCOSE International Symposium 33 (1): 851-866. https://doi.org/10.1002/iis2.13057.
- Frosch, R. A. 1969. "A new look at systems engineering." *IEEE Spectrum* 6 (9): 24–28. https://doi.org/10.1109/MSPEC.1969.5213670.
- Griffin, M. D. 2010. "How Do We Fix Systems Engineering?" Paper presented at the 61st International Astronautical Congress, Prague, CZ, 27 September – 1 October.
- INCOSE. 2025."Engineered Systems." https://www.incose.org/about-systems-engineering/system-and-se-definitions/engineered-system-definition.
- Maier, M. and E. Rechtin. 2009. *The Art of Systems Architecting*, 3rd ed. Boca Raton, US-FL: CRC Press.

- Pennotti, M., P. Brook, and D. Rousseau. 2024. "Systems engineering and the pursuit of elegance: A transdisciplinary approach to complex problems." *INSIGHT* 27 (2): 8-14. https://doi.org/10.1002/inst.12481.
- Polya, G. 1945. "How to solve it." Page references to 1957, rev. 2nd ed. Doubleday Anchor, New York. *Psychology Today*, P. (2022). Heuristics. Available: https://www.psychologytoday.com/us/basics/heuristics. Retrieved Aug. 8, 2022.
- Psychology Today. 2025. "Heuristics." https://www.psychologytoday.com/us/basics/heuristics.
- Romanycia, M. H. J., and F. F. Pelleti. 1985. "What Is a Heuristic?" *Computational Intelligence* 1: 47-58.
- Rousseau, D., M. Pennotti, and P. Brook. 2024. "The Spectrum and Evolution of Systems Engineering's Guiding Propositions." *INSIGHT* 27 (2): 27-31. https://doi.org/10.1002/inst.12484.
- Sillitto, H., J. Martin, D. McKinney, R. Griego, D. Dori, D. Krob, P. Godfrey, E. Arnold, and S. Jackson. 2019. *Systems engineering and system definitions*. San Diego, CA: INCOSE.
- Watson, M. D., B. Mesmer, and W. Farrington. 2018. "Engineering Elegant Systems: Postulates, Principles, and Hypotheses of Systems Engineering." Paper presented at the AIAA Complex Aerospace Systems Exchange 2018, Future of Systems Engineering Panel, Orlando, US-FL, 17-19 September.
- Watson, M. D., B. Mesmer, G. Roedler, D. Rousseau, R. Gold, J. Calvo-Amodio, C. Jones, W. D. Miller, D. Long, S. Lucero, R. W. Russell, A. Sedmak, and D. Verma. 2019. "Systems engineering principles and hypotheses," *INSIGHT* 21 (1): 18-28.
- Watson, M. D., B. Mesmer, G. Roedler, D. Rousseau, J. Calvo-Amodio, C. Keating, W. D. Miller, S. Lucero, R. Gold, C. Jones, D. Long, R. W. Russell, and A. Sedmak. 2022. "Systems Engineering Principles and Hypotheses," INCOSE Technical Product.

ABOUT THE AUTHORS

Javier Calvo-Amodio is chair of the INCOSE Systems Science Working Group, a fellow of the American Society for Engineering Management, an associate professor of nuclear engineering and industrial technology at the University of Idaho, a guest professor at Keio University in Japan, a visiting fellow at the Centre for Systems Studies at Hull University, UK, and deputy editor of the *Systems Research and Behavioral Sciences Journal*. At the University of Idaho, his research focuses on developing a fundamental understanding of the scientific foundations of systems as they apply to industrial engineering, systems engineering, and engineering management to enable better engineering of organizations.

Harington Lee is a systems engineering professional at Sierra Nevada Corporation where he is the deputy architect on a US Air Force program. Harington has more than 9 years of systems engineering experience across multiple defense and aerospace programs. He currently is a co-chair of the Systems Science Working Group and is leading the heuristics effort. He holds a BS in electrical engineering from the University of Illinois and an MS in systems engineering from George Washington University. He also is an INCOSE certified systems engineering professional (CSEP).

Veronika Shteingardt is a PhD student at the faculty of data and decision sciences at the Technion, Israel Institute of Technology, Haifa, Israel under the supervision of Prof. Dov Dori. She received her MSc degree in space systems engineering from the Skolkovo Institute of Science and Technology in 2016 and an MSc (2012) and Bachelor's degree (2010) from Bauman Moscow State Technical University. From 2012 to 2024, she held various engineering roles in the space and high-tech industries.

Dorothy McKinney is a seasoned systems engineer with nearly four decades of experience in the field. Based in San Jose, California, and originally from New Jersey, she spent 34 years at Lockheed Martin, where she witnessed and contributed to a significant evolution in systems engineering—from a fragmented, bureaucratic approach to a more integrated and adaptive practice. A member of INCOSE since 1992 and an INCOSE fellow, McKinney values the organization for its role in spreading the "systematic application of common sense" that defines systems engineering.

Dov Dori is a pioneer and fellow of INCOSE, life fellow of IEEE, and fellow of IAPR and AAIA. He is professor emeritus of systems engineering and head of the enterprise systems modeling laboratory at the faculty of data and decision sciences, Technion, Israel Institute of Technology. He was a visiting professor at MIT intermittently between 1999 and 2020. In 1993, Dr. Dori invented object-process methodology, OPM, which has become ISO 19450 international standard. He wrote two books on OPM, which have been central to model-based systems engineering (MBSE) and provided the basis for the MBSE edX certificate program and MOOC series. Prof. Dori has supervised over 60 graduate students and authored 400 publications, cited over 8600 times. He chaired nine international conferences and was associate editor of IEEE T-PAMI and *Systems Engineering*. He was co-founder and co-chair of the IEEE Society of Systems, Men, and Cybernetics technical committee on MBSE. He has received various research, innovation, and teaching awards, including the INCOSE pioneer award and INCOSE propeller hat award.

SEPTEMBER 2025
VOLUME 28/ ISSUE 4

On the Meaning, Purpose, and Value of Systems Engineering

Michael Pennotti, mcpennotti@icloud.com; **David Rousseau,** david.rousseau@systemsphilosophy.org; **Peter Brook,** peterbrook47@gmail.com; and **Javier Calvo-Amodio,** javier@uidaho.edu

The Bridge Team, comprising the four authors, was established in late 2020 as a project within the "Future of Systems Engineering" (FuSE) initiative to realize the *Systems Engineering Vision 2035* (INCOSE 2021). The team's charter was to develop a framework for relating the subjects of two other FuSE projects, one compiling a substantial repository of "systems engineering heuristics" and another refining a smaller number of "systems engineering principles." The framework that was developed is described in (Rousseau et al. 2025).

In the process of investigating these issues, the authors undertook a wide-ranging review not only of the diversity of published "guiding propositions" for systems engineering but also into how the practices and purposes of systems engineering emerged and have continued to evolve. The result of that investigation is the subject of this article.

The first published reference to systems engineering as a distinct discipline appeared in a paper in the *Proceedings of the Royal Society of London* in October of 1950 (Kelly 1950). In it, Mervin Kelly, who became president of The Bell Telephone Laboratories in 1951, wrote:

> *"As [communications] technology … has broadened and become more complex, the choice of the technical paths to be pursued … has become increasingly difficult. It is this situation that has led to the evolution of the systems engineering function…"*

Kelly described a Bell Labs' systems engineering organization at the same level of importance as its research and development departments. One might chuckle at Kelly's reference to the *growing complexity* of communications technology at a time when the national telecommunications network was built from copper wires, electromechanical relays, and vacuum tubes. Nevertheless, the technology in 1950 was certainly more complex than that which preceded it. This focus on the need to deal with increasing complexity was to become a recurring theme in the evolution of systems engineering.

In another of the earliest papers on systems engineering (Engstrom 1957), Elmer Engstrom, then senior executive vice president for research at Radio Corporation of America (RCA) and later to become RCA president and CEO, cited "…*the need to integrate a wide variety of novel electronic devices (e.g., radars, bomb sights, fire control systems and communications equipment) into World War II aircraft as a major impetus for the expansion of systems engineering*" during the 1940s.

Engstrom also provided an example of the successful application of systems engineering at RCA, the development of a black and white compatible color television system in the 1950s. Based on his experience with systems engineering in practice, Engstrom defined only two requirements for successful systems engineering:

- *"First, a determination of the objective that is to be reached, and*
- *"Second, a thorough consideration of all the factors that bear upon the possibility of reaching the objective and the relationships among these factors."*

The first requirement is important because most often the individual requirements of multiple diverse stakeholders are inconsistent and may even be in conflict. Systems engineers must manage tradeoffs between these disparate requirements. In doing so, they must maintain a singular point of reference that may not be compromised. The second requirement emphasizes the need for the systems engineers to understand the underlying technologies across a wide range of disciplines to ensure that the system objective is met.

The first systems engineering textbook was published by Harry Goode and Robert Machol of the University of Michigan in 1957 (Goode and Machol 1957). In their preface, the authors state:

> *"This book develops no general theory. It presents experience, the parts and pieces, and the relationships among them …."*

Goode and Machol entitled Chapter 1 of their text, "*Complexity – The Problem*." Echoing Kelly's earlier theme, they assert, "*The complexity of man's existence increases at a growing rate*." To support their point, they present numerous examples of measures that exhibit what they refer to as "*continuous and exponential growth*." They identify "*large scale systems*" as solutions developed to cope with this increasing complexity. They then describe successful applications of systems engineering in the fields of, "*communications, transportation, industry [i.e., manufacturing], commerce [i.e., logistics] and military systems*." In keeping with Engstrom's requirements, all of these examples are domain specific, and reflect a thorough understanding of both the problem to be solved and the underlying technologies required to do so. They present no specific solution techniques; they simply demonstrate the application of systems thinking to the engineering of systems.

Perhaps the greatest achievement of systems engineering during its first three decades was the Apollo lunar landing in July 1969. Throughout the decade of the 1960s, the objective to be achieved remained clear. As set forth by President John F. Kennedy in his May 1961 address to Congress (Kennedy 1961), it was:

- *"Land a man on the moon*
- *"Return him safely to the earth*
- *"Before this decade is out."*

Factors that affected the possibility of reaching this objective were many and varied, including the need for more powerful rockets, orbital mechanics, rendezvous and docking, high-speed atmospheric re-entry, human life support – both during transit and on the lunar surface – and large-scale precision manufacturing. And, of course, the relationships among these factors.

Despite the many successes of systems engineering during its first three decades, by the time of the Apollo landing, two opposing concerns had begun to emerge.

The first was, given the growing recognition of the importance of systems engineering, how an acquiring agency could evaluate and objectively compare the systems engineering proposals of competing bidders. To address this concern, the U.S. Air Force published Mil-Std 499 Systems Engineering Management in July 1969, coincidentally the same month as the successful Apollo landing (U.S. Air Force 1969). The opening paragraph of the standard stated:

"The purpose of this standard is to provide a set of criteria that will serve as a guide to:

"(a) contractors preparing Systems Engineering Management plans (SEMPs) …; and

"(b) Government personnel when either tailoring a bid work statement calling for SEMPs or competitively evaluating and validating SEMPs…"

Notice, however, that the standard shifted the focus from the objective to be achieved to the process to be followed. Further, no evidence was offered to indicate that the process described in the standard produced better systems or increased the likelihood of achieving the desired objective.

At the same time that the Air Force was calling for a more structured approach to systems engineering, the then assistant secretary of the Navy for research and development, Robert Frosch, voiced the opposing viewpoint. In a 1969 paper (Frosch 1969), Frosch, who was later to become assistant executive director for U.N. environmental programs, then the fifth NASA administrator, and finally vice president of research at General Motors, cited numerous examples in which systems engineering had failed to produce desired results. He offered his assessment of why that was so, cautioning that, "*We have lost sight of the fact that engineering is an art, not a technique; technique is a tool*." He questioned whether systems engineering was producing "*elegant solutions to real problems*," and called for bringing, "*the sense of art and excitement back into engineering*." This sentiment widely resonated with practicing systems engineers, and twenty-five years later Frosch's paper was republished as a 'classic' systems engineering paper by NASA (Frosch 1993). Unfortunately, the focus on the process to be followed rather than the objective to be achieved, that Frosch warned against in 1969, appears to have reasserted itself, and continues to this day. For example, 50 years later, ISO/IEC/IEEE 42020:2019 (ISO/IEC/IEEE 2019), which remains current, describes six processes for generating and managing architectures for software, systems and enterprises. These processes comprise 45 "required" activities and 416 "recommended" tasks!

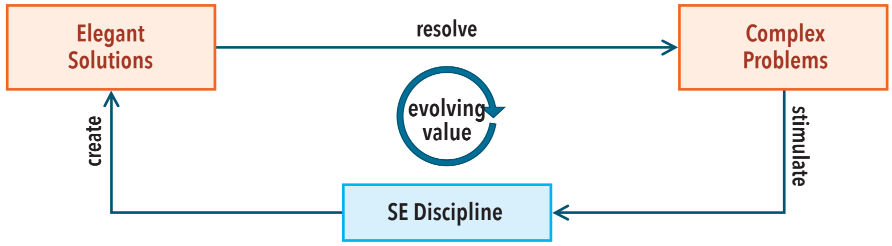


Figure 1. The "value loop" Illustrating the purpose of systems engineering

Integrating and reflecting on the insights gained from the observations above led us to see the purpose of systems engineering in terms of what we call the "value loop," illustrated here, and defended in more detail in (Pennotti et al. 2024b, 2024a). See Figure 1.

As shown in the diagram, the purpose of systems engineering is to create elegant solutions to resolve complex problems. By "complex problems" we here mean to embrace the full range of problems described as complex, from purely technical problems like those described by early pioneers like Kelly, Engstrom, Goode and Machol, that today might be referred to as complicated, to problems less amenable to systematic analysis, like those encountered in the organizational systems that create solutions to the socio-technical systems in which they are used. This is a closed loop in which the continuing increase in the complexity of the problems systems engineers seek to address stimulates continual improvement in the discipline. In this way systems engineering's value continues to evolve and increase over time.

This leaves us with the question of what makes a solution elegant. In a 2010 address in Prague, former NASA administrator Michael Griffin proposed four criteria that an elegant solution must satisfy at a minimum (Griffin 2010). It must:

- *Work as intended* — The system must successfully resolve the complex problem.
- *Be robust* — If circumstances change from initial assumptions, the system must degrade gracefully, rather than fail catastrophically.
- *Be efficient* — not only in terms of financial resources, but also human resources, energy resources, environmental resources, etc.
- *Minimize unintended consequences.*

Griffin considered these criteria necessary but not necessarily sufficient. They do, however, represent a reasonable starting point for a conversation about the meaning of the term "elegance" in the context of systems engineering problem-solving. Since

Griffin's proposal, a number of others have added their voices to the elegance conversation, including (Efatmaneshnik and Ryan 2019, Madni 2018, Salado and Nilchiani 2013, Watson et al. 2020).

To ensure systems engineering continues to evolve to meet the increasingly complex challenges of the future, we offer the following recommendations to the systems engineering community:

- **Put the engineering back into systems engineering** — because "a thorough consideration of all the factors that bear upon the possibility of reaching the desired objective and the relationships among these factors" require an understanding of the underlying technologies.
- **Reinforce and encourage the "art" of systems engineering** — because systems engineers must make decisions about what to build and why and resolve the tension between desirable attributes that may be in conflict, for example robustness and efficiency or adaptability and resilience.
- **Require that all proposed systems engineering techniques, processes, and tools be validated with evidence of their ability to aid in achieving system objectives** — because the tools we build for ourselves must be held to the same standards as the systems we design for others.
- **Continue to expand our understanding of elegant solutions and how to achieve them** — because the quest for elegant solutions stimulates innovation by helping us move from "either-or" to "both-and" solutions. Examples of this from other domains include elegant solutions to engineering, economic, political, communications, and mathematical problems.
- **Develop case studies for recent system successes to understand the extent to which they represent elegant solutions and if so, why, and how that was achieved**. Examples might include the James Webb Space Telescope (344 single points of failure), the Mars Rover Landings (7-minutes of terror), Terminal 5 at Heathrow Airport, SpaceX Falcon 9 booster recovery, mobile communication ecosystems (e.g., Apple, Google, and Starlink) and, no doubt, many other less widely recognized systems.

These actions will help guide systems engineering's evolution towards becoming a discipline more grounded in principles than formal processes and assist systems engineers transition to being more focused on innovation and creativity than checklists. This transition will not only resolve the concerns about the character of systems engineering raised by Frosh, Griffin, and others, but also allow systems engineering to reclaim its original meaning, fulfill its enduring purpose and assure its continuing value in an increasingly complex world. ■

Taken from a presentation to the INCOSE Fellows at their semiannual meeting on February 3, 2025, in Seville, Spain.

REFERENCES

- Efatmaneshnik, M., and M. J. Ryan. 2019. "On the Definitions of Sufficiency and Elegance in Systems Design." *IEEE Systems Journal 13* (3): 2077–2088. https://doi.org/10.1109/JSYST.2018.2875152.
- Engstrom, E. W. 1957. "Systems Engineering: A Growing Concept." *Electrical Engineering* 76 (2): 113–116.
- Frosch, R. A. 1969. "A new look at systems engineering." *IEEE Spectrum* 6 (9): 24–28. https://doi.org/10.1109/MSPEC.1969.5213670.
- Frosch, R. A. 1993. A Classic Look at Systems Engineering. Speech to IEEE Group on Aerospace and Electronic Systems, IEEE International Convention, New York, NY-US, 26 March 1969. Reprinted. In T. Hoban and W. M. Lawbaugh (Eds.), *Readings in Systems Engineering* SP-6102: 1–7. NASA Scientific and Technical Information Program.
- Goode, H. H., and R. E. Machol. 1957. *System Engineering: An Introduction to the Design of Large-scale Systems*. McGraw-Hill.
- Griffin, M. D. 2010. "How Do We Fix System Engineering?" *61st International Astronautical Congress*, Prague, CZ, 27 September – 1 October, 27.
- ISO/IEC/IEEE. 2019. ISO/IEC/IEEE 42020:2019 Software, Systems and Enterprise Architecture Processes.
- Kelly, M. J. 1950. "Bell Telephone Laboratories—An example of an Institute of Creative Technology." *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 203 (1074): 287–301.
- Kennedy, J. F. 1961. Special Message to Congress on Urgent National Needs. Delivered in Person before a Joint Session of Congress.
- Madni, A. M. 2018. "Design Elegance and Systems Engineering." In *Transdisciplinary Systems Engineering*, 119–132: Springer.
- Pennotti, M., P. Brook, and D. Rousseau. 2024a. "Systems Engineering and the Pursuit of Elegance: A Transdisciplinary Approach to Complex Problems." *INSIGHT* 27 (2): 8–14. https://doi.org/10.1002/inst.12481.
- Pennotti, M., P. Brook, and D. Rousseau. 2024b. "The Evolution of Systems Engineering as a Transdiscipline." *Systems Engineering* 27 (5): 1–12. https://doi.org/10.1002/sys.21757.
- Rousseau, D., M. Pennotti, M., and P. Brook. 2025. "Spectrum and Evolution of Systems Engineering's Guiding Propositions." *Systems Engineering* n/a (n/a). https://doi.org/10.1002/sys.21809.
- Salado, A., and R. Nilchiani. 2013. "Using Maslow's hierarchy of needs to define elegance in system architecture." *Procedia Computer Science* 16: 927–936. https://doi.org/10.1016/j.procs.2013.01.097.
- U.S. Air Force. 1969. MIL-STD 499: *Systems Engineering Management*. US Department of Defense.
- Watson, M., B. Mesmer, and P. Farrington. 2020. *Engineering Elegant Systems: Theory of Systems Engineering* Vol. 20205003644.

ABOUT THE AUTHORS

Michael Pennotti is an INCOSE fellow and was formerly a distinguished service professor in the school of Systems and Enterprises at Stevens Institute of Technology, Hoboken, NJ, USA. He served as associate dean for academics in the school and later, director of systems and software programs. Prior to joining Stevens in 2001, Mike spent 20 years as a systems engineering practitioner and leader at Bell Laboratories, designing and improving the operational performance of four generations of undersea surveillance systems for the U.S. Navy, and another 10 years applying systems principles to enterprise systems in executive positions with AT&T, Lucent Technologies, and Avaya. He is a senior life member of the IEEE and was a co-founder and coach of the INCOSE technical leadership institute. He holds PhD and MS degrees in electrical engineering from the Polytechnic Institute of New York, a BEE from Manhattan College, and is a graduate of the AEA/Stanford Executive Institute for technology executives.

> continued on page 69

Accelerate Smart Product Innovation with MBSE for Electronics

SEPTEMBER 2025
VOLUME 28/ ISSUE 4

Principles for Minimizing Unintended Consequences

David Rousseau, david.rousseau@systemsphilosophy.org; and **Julie Billingham**

■ **ABSTRACT**
A significant challenge to the success of systems engineering solutions is the risk of unintended consequences. This has traditionally been considered to be a real but unactionable requirement. Here, we analyse the notion of unintended consequences and propose an equivalent but actionable requirement, which we relate to the concept of 'harmony'. The implication of this notion is that in order to assure solution success it has to be architected in concert with considering the structure and dynamics of the system of systems in which it will be deployed. We conducted a study of natural ecosystems as a case study from which we could glean relevant architecting principles. From this we developed a general model of the structure and dynamics of a complex system of systems. From this study and model we distilled a set of general principles for systems architecting. Along the way we introduced a framework for understanding the relationships between the notions elegant, complex, complicated, and simple. We also introduce a new perspective on emergence at the level of parts, supplementing classical notions of emergence at the level of the whole.

■ **KEYWORDS:** unintended consequences, elegant solutions, organisation in biology, community ecology, general architectural principles, structure and dynamics of complex systems of systems, evolution of elegance

1. INTRODUCTION

An important paradigm emerging from the "Future of Systems Engineering" (FuSE) program to realize the *Systems Engineering Vision 2035* (INCOSE 2021) is the idea that "the purpose of systems engineering is to create elegant solutions to complex problems" (Pennotti et al. 2024a, 2024b, 2025). The origin of this paradigm goes back to former NASA Administrator Robert Frosch, who was concerned about high failure rates in systems engineering projects. In 1969, he suggested the maxim (hence forward, Frosch's elegance maxim) that systems engineers should seek to deliver "elegant solutions to real problems" (Frosch 1969, 1993). He pointed out that this requires creative talent in addition to the diligent application of tools, methods, and processes. This conception was endorsed and expanded by another former NASA Administrator, Michael Griffin, who in 2010 proposed that in the context of systems engineering an 'elegant' solution is one that works as intended, is robust, is efficient, and has minimal unintended consequences (Griffin 2010) (hence forward, Griffin's elegance formula). Griffin proposed this formula as a minimal requirement for attaining 'elegance', while calling for the formulation to be expanded and refined. Many have taken up the call, for example (Efatmaneshnik and Ryan 2019; Iandoli et al. 2018; Madni 2012, 2018; Rousseau et al. 2019; Salado and Nilchiani 2013; Watson et al. 2014, 2019, 2020; Watson and Griffin 2014) . We will not review these works here but point out three key developments that we will draw on in the present paper.

First, there has been a shift to viewing Griffin's elegance formula as not providing specific parameters but categories of desirable characteristics, so criteria like 'robustness' should be interpreted loosely and broadly, to sweep in related ilities such as resilience, reliability, graceful degradation, and maintainability (Rousseau et al. 2025, Watson and Griffin 2014). This makes the further development of the formula much more manageable, because it avoids having to establish a precise common agreement about what is included in each criterion. However, it also opens the door to recategorizing the diverse range of criteria, so the categories could perhaps, from a contemporary perspective (15 years after Griffin, himself 40 years after Frosch), be more nuanced. For example, we could suggest Griffin's elegance criteria be recategorized as functionality, dependability, affordability and, of course, minimising unintended consequences.

Second, the *Systems Engineering Vision 2025* (INCOSE 2014), identified that the risk of project failure scales with the complexity of problems addressed and solutions offered. It noted that complexity is rising exponentially, especially in the light of Industry 4.0 and Society 5.0. In the light of this, the FuSE "Bridge Team" proposed a modified Frosh's elegance maxim, namely that "the purpose of systems engineering is to deliver elegant solutions to complex problems" (Pennotti et al. 2024a, 2024b, 2025). This adjusted maxim connects

Frosch's and Griffin's responses to the long-standing problem of engineering failures due to over-reliance on processes and tools to contemporary concerns about rising complexity as aggravating the risk of solution failure.

Third, the principles and methods for achieving elegant solutions are still fairly underdeveloped. In response, the Bridge Team recently proposed several strategies to address this. This included calling for studies into successful elegant solutions in multiple domains, in order to uncover practical principles relevant to architecting and designing elegant solutions (Pennotti et al. 2025). This encourages engineering researchers to also learn from successes, and not just case studies about failures. In particular, the Bridge Team recommended studies into solutions where the solution engineers did not self-identify as systems engineers, and so their approach might not have mirrored established systems engineering practice. Such studies could hold valuable lessons for improving systems engineering.

Our present paper is a response to that last mentioned call. Our aim is to support and advance the search for principles that would aid in delivering elegant solutions.

However, as we see it, there are two significant unresolved issues that impede the quest for such principles. The first is that the final criterion in Griffin's elegance formula is defined negatively (it defines what should *not* happen), and hence it seems unactionable. The second is that Frosch's elegance maxim suggests that the elegance of a solution is established in relation to the complexity of the problem it resolves. At present, we have no framework for analysing this relationship, which makes the adequacy of the solution un-assessable.

Our paper begins by proposing our solutions to both these issues. With these solutions in hand we then undertook a study of complex natural ecosystems to uncover principles underpinning their enduring elegance across a range of scales. We present our findings here, and we generalise the principles we discovered to show their relevance to systems architecting and design across application domains.

Finally, we will argue that we have uncovered a set of general principles for systems architecting relevant to all projects, and that they usefully supplement the specialised architecting principles that every particular project derives from project-specific early-stage activities such as requirements gathering and stakeholder analysis.

2. PREPARATORY CONCEPTUAL AND SCIENTIFIC FOUNDATIONS FOR OUR STUDY

2.1 Making the elegance formula fully actionable

Griffin's proposal for elegance categories in the context of systems engineering was that an 'elegant' solution is one that works as intended, is robust, is efficient, and has minimal unintended consequences (Griffin 2010). However, the last criterion is unactionable, as it does not provide a measure for evaluating the degree to which this criterion is accommodated in the design. Directing the engineer to design a solution that has minimal unintended consequences is to require the engineer to take account of events and phenomena that are unforeseeable at the point of design. If these were foreseeable, the design would have been adjusted to accommodate them. Addressing this conundrum is the purpose of the present sub-section.

This is an important matter, because minimising unintended consequences seems to be the key to solving the problem of solution failure. Griffin in fact expressed just this view, saying:

> "Complex systems usually come to grief, when they do, not because they fail to accomplish their nominal purpose. While exceptions certainly exist, it remains true that almost all systems which proceed past the preliminary design phase will, in fact, accomplish the tasks for which they were explicitly designed. Complex systems typically fail because of the unintended consequences of their design, the things they do that were not intended to be done" (Griffin 2007).

What is meant by failure here is that the solution system cannot enduringly perform its intended function, or cannot do so dependably, or affordably. Moreover, the intimation is that this degradation of functionality or ilities performance of the solution system comes about due to unintended consequences of the system in operation. The most natural reading of this is to say the solution system's actions have negative impacts on other systems in its operational environment, systems its actions were meant to be at least compatible with, if not actually supportive of. The implication is that these negative impacts degrade the functional and ilities performance of allied systems. A natural response from these allied systems would be adaptations to minimise or remove the negative impacts on them. Such adaptations could be changes in their behaviour, structure, or environment. This means that the solution system is increasingly operating in a context different from what it was designed for. These changes would cumulatively undermine the functional or ilities performance of the solution system, potentially up to the point of solution failure.

We can now see a qualitative difference between the first three criteria in the Griffin elegance formula and the last one: the first three are concerned with desirable attributes of a good solution (Rousseau et al. 2019), while the last category points to the risk of the first three becoming degraded in operation. Our objective now is to change this so that the last criterion also expresses a positive property of a good solution.

From the arguments developed above we can now conclude that the risk of solution failure resides in conflicts between the interests of the solution system and other systems in its environment, systems it was *not* designed to have negative relations with. We can characterise this as that the actions of the solution produce 'friction' between the relevant parties (i.e., parts in the system of allied systems), and that this friction can increase to the point of causing solution failure (this notion of 'friction' was introduced by the military theorist and philosopher of war Carl von Clausewitz (Clausewitz 2008)).

Designing a system that will never trigger significant friction is clearly impossible given the uncertainty of the future. In this light, an elegant solution will be one that includes provision for monitoring the emergence of sources of friction and means for adaptation to mitigate or eliminate them. In short, the solution system should have features that enable it to ongoingly operate "in harmony" with the systems it was intended to be compatible with. For reasons that will become clear later on, this to-be-aimed-for outcome is more aptly characterised as "external harmony," or, more precisely, harmony between systems loosely coupled into a system of systems. In systems science there is a technical term for this condition of harmony between networked systems, namely "syntony."

The requirement for the solution system to be harmonious with what might be called 'allied systems,' is eminently actionable, in the sense that it requires the systems architect to consider the impact of the planned solution on the ability of allied systems to function as *they* were intended to do.

We thus now propose to reframe the final category of the elegance formula as a category we will call "harmony," a concept we will further refine later on. The elegance formula thus now covers functionality, robustness, efficiency and harmony, and we could say that an elegant solution is one that is effective, robust, efficient and harmonious, it being understood that each of these labels embraces a range of related factors.

This suggests that a system's operational milieu can be conceptualised as something like an ecology of diverse interacting systems. An inference we can draw from this argument is that if we would like to uncover principles for designing elegant solutions then we would do well do study examples of complex systems of systems that have endured over substantial timescales, in order to try an understand the principles by which they were able to individually and collectively avoid being undermined by unforeseeable events.

2.2 Relating elegance to complexity

Frosch's revised elegance maxim states that "the purpose of systems engineering is to deliver elegant solutions to complex problems". This is a fine tenet, but on what basis could we judge that this has been achieved? The terms 'elegant' and complex' can have different meanings in different contexts, but they always refer to a scale – problems can be more or less complex, and solutions can be more or less elegant. This suggests that there must be some kind of optimality or sufficiency principle for relating the degree of elegance of the solution to the extent of the complexity of the problem it is meant to solve. Such a principle would enable us to judge a specific solution to be elegant (or not) *in relation to* the complexity of the specific problem it aims to solve.

We will now present a perspective on complexity from which can uncover such an 'optimal complexity principle' for elegant solutions.

The term 'complex' has many meanings tailored to different contexts, and we will not review or critique them here, but we would say that we are *not* advocating the establishment of a universally applicable definition. However, for the present case we will propose a view on complexity that we think is appropriate to the present context, one that reflects classical perspectives and also reflects uses in everyday language.

For present purposes, we propose that a system is complex if it contains many kinds of parts, many kinds of inter-part relationships, and hence can be in many different states, and can display many kinds of behaviours (Francois 2004, p. 103 ff). Such a situation typically makes the system's behaviour hard to describe, model, explain, or predict. This entails that complexity has both a metaphysical dimension (referring to the inherent nature of the complex system) and an epistemological one, focussed on challenges to our ability to analyse and understand the system (Alhadeff-Jones 2008, Estrada 2024, Heylighen et al. 2006, INCOSE Complex Systems Working Group 2015). For present purposes we will focus on the metaphysical dimension. To us, what is objective and intrinsic to the complex system itself is its internal variety in terms of parts, inter-part relationships, potential states, and state-changes (behaviours). On this measure, systems are more complex if they have more actual or potential internal variety. This establishes a simple scale for judging complexity.

In contrast, we can say that something is 'simple' if has few kinds of parts, and hence few inter-part relationships, potential states, and potential behaviours. Hence, for simple systems it is relatively easy to describe, explain, and predict their behaviours.

All systems have behaviours, that is, they change on the basis of the interplay between environmental influences and their internal organization. For complex purposeful systems, such as organisms and engineered systems, their functional and ilities performance is dependent on whether their design, expressed as internal complexity, lives up to the complexity of the challenges they face.

Obviously, the more complex the challenges to be met, the more complex the solution system will have to be. However, complexity in the problem does not directly translate into complexity in the solution. Sometimes, a solution to a complex problem can be relatively simple.

The bifurcated needle used for smallpox vaccination is a prime example of a simple solution to a complex operational problem. It was effective, cheap to make, simple to use, easy to prepare for re-use, easy to teach, robust under a wide range of conditions, and a key innovation towards achieving the goal of eradicating smallpox on a global scale (Jenkins 2017). This solution is widely regarded as an *elegant* solution exactly because it was sufficiently complex to solve the problem but no more complex than it needed to be.

Michael Pennotti has expressed a phrasing of this saying that an engineered system should be made as simple as possible, but no simpler [personal communication, 01 May 2024]. As Pennotti pointed out, this echoes for engineering an aphorism about theory building widely attributed to Einstein, with precursors in Occam and Aristotle.

This seems to us to state a general principle about elegant solutions – if an offered 'solution' is less complex (or simpler) than is needed to solve the problem, it is not a solution. If it is more complex (or less simple) than it needs to be to solve the problem, then it could be a solution, but it cannot be an elegant one.

This gives us the 'optimal complexity principle' we sought: an elegant solution is one that attains the minimal complexity needed to address the complexity of the problem.

We note that the proposed optimal complexity principle is compatible with Griffin's elegance formula. A design that fails to satisfy this principle cannot meet all the criteria of Griffin's elegance formula: it might be effective, but it would be unlikely to meet ilities requirements such as efficiency. We also note that the elegance of a solution is an emergent property of the relationship between the solution and the problem, and not an intrinsic property of the solution system. In practice when a solution is called 'elegant' that judgement is often based on unexpressed assumptions about the nature of the problem that it was meant to solve. Making this conditionality explicit might help an architect choose between alternative possible 'solutions.'

From this we can see that, given the problem, an elegant solution can be anywhere on the spectrum from very simple to very complex. But what of systems that are more complex than necessary? We would call them 'complicated'. This is close to the ordinary language meaning of complicated, for example someone might say "now you are just complicating things" in response to a proposed strategy or explanation, meaning that the strategy or explanation includes items or processes not needed to get to the conclusion or desired outcome.

Extreme examples of systems that are 'complicated' in this sense are to be found in the well-known Rube Goldberg machines. Goldberg is famous for his cartoons depicting "complicated gadgets performing simple tasks in indirect, convoluted ways" ('Rube Goldberg' 2025).

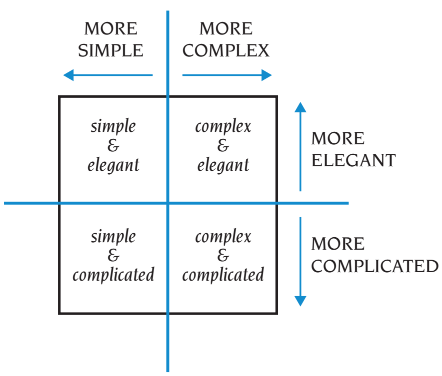


Figure 1. Balancing simplicity and complexity against elegance and complicatedness

So now we have four conditions: systems can be simple (having low internal variety) or complex (having high internal variety) and simultaneously could be either elegant (exhibiting optimal complexity) or complicated (exhibiting excessive complexity). We can represent this tension abstractly as in Figure 1, designating the average types in each case. Reality is of course more nuanced, and so in practice we have a spectrum of designs that reflect degrees of actuality on the simplicity vs complexity spectrum, and degrees of achievement on the elegance vs complicatedness spectrum. However, we can now see that in general terms 'complex' is the opposite of *simple*, and 'complicated' is the opposite of *elegant*.

Complicated systems cannot be elegant, because the extra complexity carries avoidable costs in development, manufacturing, and sustainment, and creates avoidable risks to all aspects of performance (functionality, ilities, and syntony). This implies that to be elegant a solution must minimise its internal variety, containing only what is essential and ensuring that these interoperate smoothly. If this is not achieved, we have internal sources of friction. So, an aim is for the solution system is to have "internal harmony." For this condition, systems science has a technical term, "concinnity". It is the flipside of the external harmony (syntony) discussed earlier.

We can now refine the fully actionable Griffin elegance formula from the previous section to say that the 'harmony' we proposed as the fourth criterion includes both internal and external harmony.

Using the optimal complexity principle inferred above we can now also paraphrase the Frosch elegance maxim, to say that the purpose of systems engineering is to deliver solutions that are sufficiently complex, but no more complex, than needed to solve a given complex problem.

3. SELECTING OUR CASE STUDY SYSTEM

As we argued earlier, if we wish to discover principles for minimising unintended consequences, we have to study examples of complex systems of systems that have been successful over a long period. For this project, we chose to study examples of enduring systems of systems in nature. Natural systems evolve over very long periods of time, a design method that we cannot emulate. However, the successes of biomimicry and genetic engineering show that if we understand the principles underlying a natural design, we can utilize those insights without being bound by nature's evolutionary timescales.

In particular, we wished to study natural ecosystems and communities, i.e., groups of different species that occur together and interact in space and time. This contrasts with the traditional ways in which engineers seek inspiration from nature, such as biomimicry. In the latter approach, individual species are studied to discover the principles behind nature's point solutions to engineering problems, e.g., strong but lightweight spider silk, tough crocodile armour plates, or high-grip limpet glue. Nature contains many examples of such elegant solutions that provide a refined capability at low cost with high resilience and sustainability. In our case, however, we wished in particular to understand durability in highly diverse loosely coupled systems of systems, to uncover and generalise the architectural principles underlying their endurability.

It was our view this could be a more productive approach than studying successful artefactual systems. Nature is rich in complex systems, from organisms to ecologies, and we see in them features that we admire from an engineering perspective, such as efficiency, resilience, and sustainability, but also features that surprise us, such as high diversity and increasing complexity. In engineered systems and engineered systems of systems, diversity and complexity typically increases fragility and risk of failure. However, nature seems not to be limited in the same way; there are highly diverse and complex ecosystems such as rainforests that are efficient, resilient, and enduringly viable under natural conditions. This is the opposite of the experience we see in the systems engineering community, and so it seems clear that that there are important engineering principles missing from the current systems engineering knowledge base. If we could understand those natural engineering principles, this would be a great boon to engineers designing solutions to complex problems that arise in complex contexts.

It may seem strange, at first thought, to compare an engineered system of systems with an ecosystem. We are used to thinking of ecosystems in terms of predator-prey interactions and competition for resources. In fact, this has been the dominant mindset in ecology until relatively recently. Now, however, there is an emerging awareness of the significant impact that beneficial interactions have in inter-species networks. Such interactions range from facilitation, such as cleaner fish that remove parasites from other fish to deep mutualisms such as exist between plants and the mycorrhizal fungi with which they trade nutrients. As we will show in the next section, these interactions support the elegance of the ecosystem as a whole.

To us, the closest analogies in nature to engineered systems of systems are found in the microbial world. Microbial communities are extraordinarily resilient and adaptable, despite being comprised of a wide variety of kinds of microbes having diverse functions and with many kinds of interactions between them. Microbes are single celled creatures that effectively function as recycling and assembly lines, breaking down complex chemicals into smaller components, either extracting energy or reassembling the parts into chemicals and structures they need, such as enzymes, proteins, biofilm, or signalling molecules. Microbial communities are characterised by cross-feeding, a process where one species makes use of the waste products of another's metabolism, and so on through a highly complex network of resources and consumer-producers, to generate a product that no one member could produce on their own. Gralka (Gralka et al. 2020) has described such communities as a distributed metabolic system.

We are confident that the principles underlying the endurability of these communities are discoverable, because, as systems scientists have pointed out, nature exhibits patterns that recur homeomorphically at different scales and with different compositions and development histories. For example, spiral forms can be seen in galaxies, tornados, flowers, and seashells. This implies that the same principles are involved in establishing enduring structures in these different contexts.

Many specialised systems sciences have arisen around the discovery of the principles, mechanisms, and processes underlying such patterns, for example control theory, hierarchy theory, communication systems theory, game theory, automata theory, etc. However, these focus on point solutions to complex systems problems.

We hold that the next frontier in engineering will be to study the principles underlying the success of network solutions to complex system-of-system problems. Perhaps in the future a new discipline called ecomimicry will complement the already productive discipline of biomimicry.

4. THE EMERGENCE OF ELEGANCE IN ECOSYSTEMS

In this section, we first present a brief orientation about natural systems, showing how engineering concepts such as function and elegance map onto ecosystems at both system and system-of-systems level. We then examine the ecosystem design architectures that are relevant for ensuring their enduring viability and show how these contribute to increasing elegance of the ecosystem. In the next section we then generalise what we learned from nature into a general model of a system of systems, from which we can derive general principles for architecting ones that will endure.

4.1 Principles underpinning ecosystem change

Life on Earth is extraordinarily diverse. We have an enormous variety of species at every scale, interacting in multiple ways, whether competing with, preying upon, or providing useful services to one another. Each individual organism is part of a population made up of members of its species. It will also be part of a community comprising the other species with which it interacts. Such a community, together with the physical terrain and non-living substances they use or produce, forms an ecosystem.

An ecosystem is thus a loosely coupled system of systems, where the component systems are mostly tightly coupled. The world's ecosystems are constantly changing, driven by biogeological cycles such as weather and tides, as well as the dynamic responses of living things to the changes these bring about. That said, there is order. The diversity and relative abundance of species varies in regular ways across space, time and scale, and there is pattern and structure in these distributions, even when they are changing. Community ecologists focus on identifying these patterns and understanding how they are generated and maintained in space and time (Mittelbach and McGill 2019, p. 1).

Within normal parameters, natural ecosystems display remarkable tolerance to change, either with a damped recovery from perturbation or a gradual shift to a new state. All this regularity points to the existence of principles that underpin complex processes and mechanisms leading to an emergent resilience and adaptability. In this section we highlight some of these principles that we have inferred from the observations of biologists.

4.2 The impetus for individuals to improve their elegance

An organism is an agentic system and can be considered to perform a range of functions comprising the things that it does, e.g. find food, eat it, grow, communicate, learn, play, reproduce, teach etc. However, it exists in a risky environment and so has to ensure its own survival while performing those functions.

Every activity in nature involves investing to generate a return. At the most basic level, a cell metabolising glucose needs to invest energy to kickstart the chemical process that ends up generating an energy return in excess of the investment. A tiger needs to invest energy to chase its prey; too many failures can weaken it to the point where it can no longer even make the attempt. There is no free lunch in nature.

It is therefore crucial for an organism to use every opportunity to reduce the energy cost of everything it does. This equates to becoming better at utilising its resources, accumulating reserves and improving the effectiveness of its functions, while limiting risk to those capabilities. In other words, it constantly seeks to improve its systemic elegance.

In this context, it is significant that the environment is changing continuously. Natural cycles cause fluctuations in the availability of resources, and many biochemical processes are sensitive to factors such as temperature. Each change has the potential to either compromise or enhance an organism's efficiency or functional effectiveness. This will stimulate it to take continuous steps to either mitigate the loss or make use of the opportunity as appropriate.

The decisions that individuals have to make while performing their functions embed the potential for making such trade-offs. There are three main strategies an organism can apply:

- **Adjust its behaviour.** An organism can choose to take different actions towards the same outcome. For example, glucose can be metabolised using either fermentation or oxidation. Oxidation generates more energy per investment, so microbes capable of both will generally choose oxidation if oxygen is present. However, fermentation is faster, so when glucose is in short supply and competition is fierce, a microbe may switch methods to increase its yield. A predator such as a tiger will prioritise hunting weak or sick prey, to increase its likelihood of success, reduce its energy investment, and reduce its risk of being injured itself. Once successful, it will mitigate the risk of a competitor stealing its catch, by eating the parts of the prey in order of nutrient value: the energy-dense fat first, so that it can catch new prey if necessary; then the organs, which are repositories of minerals and synthesised vitamins; and lastly the muscle, which has nutritional value but takes the most energy to digest.
- **Adjust its niche.** An organism can move to a new niche or transform the one it is in to make it more suitable. Beavers build dams to provide protection from predators, elephants dig waterholes in dry riverbeds, and microbes build biofilm to create micro-niches and store resources.
- **Change its own body plan.** Over multiple generations, a population of organisms may evolve a changed body plan that reduces costs or adds new functions. This can happen in relatively short timescales through e.g., selection for better camouflage, epigenetic switching to reduce the size of young when resources are limited, or, for microbes, the transfer of genes directly from one species to another. Unnecessary functions fall away, such as eyes in deep-cave fish. We can see the results of this longer-term improvement program by looking at nature's design successes. Biomimicry already replicates many of these excellent solution architectures in engineering, as discussed earlier.

Using these approaches, individuals can incrementally improve their elegance in general, as well as responding to friction or opportunities generated by environmental fluctuations or changes in their competitive landscape.

4.3 The role of a community in increasing individual elegance

So far, we have spoken about individual organisms as though living in isolation, but in fact, self-interest can be well served by being part of a multi-species community. Community members are connected through a network of interactions, and this network provides benefits that can be leveraged by an individual to improve its own elegance.

Species interactions in communities and ecosystems have traditionally been viewed through the lens of antagonistic relationships such as competition and predation. It is only in relatively recent times that biologists have started to pay more attention to beneficial interactions. This, despite the fact that beneficial interactions are extremely common in nature, have been well known since ancient times and are generally fascinating (Mittelbach and McGill 2019, p. 158). Such interactions include mutualisms, in which both parties benefit (e.g., flower pollination by insects, seed distribution by frugivores), and facilitation, where one species alters the shared environment in a way that is beneficial for others (e.g., beaver dams, coral reefs). Interactions may, in fact, be context dependent, e.g., they may switch between beneficial and antagonistic in different seasons or life stages.

The significance of beneficial interactions can be seen in the concept of an ecological niche. An organism's *fundamental niche* is the habitat or environment it is capable of occupying in the absence of interactions with other species. Its *realized niche* is the environment it actually occupies in the presence of interacting species. Early models and lab experiments suggested that competition and predation would ensure that the realized niche was always smaller than the fundamental one. However, we now know that mutualisms can support a species in a previously unsuitable environment, leading to a larger realized niche. This outcome is more consistent with observed nature.

An intriguing example of the above occurs in the microbial community kefir, which is used to ferment milk into the yoghurt-like drink of the same name. This highly diverse group consists of a complex network of ~50 species of bacteria and yeasts, with many feeding off the byproducts of others' metabolisms. The dominant bacterium in this community cannot survive in milk on its own; it needs the metabolic contributions of the others. Intriguingly, it has not been found anywhere else (Blasche et al. 2021).

Using a byproduct of another organism's activities is one of the most common beneficial interactions. Being in close proximity to the producer enables the organism to reduce the cost of finding those resources. Dung beetles clear animal waste, while frugivores eat the fruit of trees. The producer benefits from waste removal and seed dispersal respectively. Sometimes, the interactions may be more chain-like than reciprocal. Plants produce toxins in their leaves designed to repel or combat the particular pathogens, parasites, or predators to which they are vulnerable. Animals suffering from disease caused by one of those pathogens may consume those leaves as medication for themselves. In kefir, members of the community not only contribute food for others, they consume substances that are inhibiting others' metabolisms, and they synthesise a range of shared products such as acidity regulators, enzymes, antibiotics, antibiotic neutralisers, and signalling molecules.

A key opportunity for increased elegance arises in association with creatures that modify the environment in significant ways. Coral and beavers are referred to as 'keystone species' because their reefs and dams form habitats that can underpin an entire ecosystem. By aggregating many species in close proximity, they also enable other types of interactions to become more efficient. The dominant kefir bacterium mentioned above is a keystone species in its community because it builds the scaffolding that supports biofilm made by others. Biofilm has a highly complex structure that one could arguably liken to a city, with transport routes, resource stores, barriers to entry, and a variety of micro-niches. In recent work by one of us (Billingham), biofilm was seen to provide the opportunity for integral feedback in the control of the kefir system, thus supporting the resilience of the community (Billingham 2024).

In the pursuit of cost-effectiveness, organisms will, over generations, shed functions that other ecosystem members can provide. Maintaining a function takes energy, even when it is unused, so if another member of a community can perform that function more efficiently, the species may ultimately evolve to lose that capability completely. This is particularly common in microbial communities, as in the kefir example above. The result in this case would be increasing specialisation of the species.

Specialisation introduces functional differentiation, an important characteristic in communities. (Hunt and Colasanti (2021) have shown that in plant communities, the minimal requirement for biodiversity to be maintained is the combination of functional differentiation, spatial heterogeneity, and the opportunity for new species to invade. Similarly, in kefir, functional differentiation drives the cross-feeding mentioned earlier, which, together with the biofilm's micro-niches, underpins its resilience.

Specialisation carries risk, however, as it can introduce fragility for an individual. Some orchids and their pollinators are so perfectly adapted to one another that neither can survive without the other. Such partners can be severely threatened by environmental changes. This probably explains why the majority of organisms fall somewhere on the spectrum between specialist and generalist.

It is worth highlighting that a population is a special type of ecosystem, being comprised of related individuals of the same species. Populations are force multipliers for the functions they have. Meerkats will take turns on watch while others forage, and geese migrating will take turns at the head of the V, allowing others to benefit from their slipstream. Both actions support efficiency for all the individuals in the population. Murmurations of starlings or balling schools of fish confuse predators, reducing risk to individuals.

Natural populations contain kinds of systems that are more complex than any that systems engineers have yet tried to create. We see this as a further potential benefit to be gained from studying natural systems. For example, in nature we can find systems that can metamorphose (e.g., caterpillars turning into butterflies), and systems of systems where the members can change from being competitive individuals to being a co-operative team to becoming a closely coupled new structure. For example, social amoebae compete for food in a pond, but if the pond is drying out, they can combine to form a 'slug' that can move across land in search for a new puddle. If it finds water the slug is disassembled into competing individuals. If water is not found, the slug is reconfigured to become a 'fruiting body' that can release spores into the wind. If a spore lands in water, it develops into a new amoeba, that can divide iteratively to form a new community of competitive individuals.

Over time, inter-species partnerships can be refined and individual elegance incrementally improved using the strategies and methods described in the previous section. As before, these changes potentially trigger friction for others, who change their own behaviour, structure, or environment, thus triggering others in never-ending waves of elegance improvement. For convenience, we will talk here about organisms changing their own structure. This is shorthand for an evolutionary process whereby, over time, individuals in a population acquire a new capability through selection.

Ultimately, the actions of individuals aim to make themselves more efficient and effective as individuals. We now turn to considering the effect of these collective efforts on the ecosystem as a whole.

4.4 Function, value, and elegance in individuals and ecosystems

Once we start thinking about an ecosystem as a whole, it is tempting to think about it as an entity with its own properties, functions, and causal powers. This can be misleading. As a loosely coupled system, an ecosystem does have emergent properties such as resilience, but it does not have causal powers itself. The power of agency–to make decisions and do things–resides in the individual organisms themselves, i.e., the closely coupled systems that are parts in the loosely coupled system of systems.

We have discussed that an organism has a range of functions, and that it is incentivised for these to be executed as efficiently and reliably as possible. This incentive arises because the environment is constantly changing, so to survive, it needs to take every opportunity to conserve resources and build reserves. It does this by improving the efficiency and effectiveness of its functions. In addition, it needs to manage risk to those qualities.

What, then, is the function of an ecosystem? Biologists have a range of views on this, depending on the objective of their analysis. Function may be framed in terms of the sum total of an impact of the members, e.g., total biomass produced, or resources consumed. In bioengineering, a microbial community might be seen as a mechanism that converts its inputs into the desired output. Some ecosystems are seen to function as regulators for an important process such as the carbon cycle, or as providing pollination services for crops. While valid perspectives, these notions of function seem rather human-centric to us and fail to reflect a self-reinforcing value that enables the ecosystem to persist.

To explore the function concept further, consider that an ecosystem is a network of interacting organisms that arises naturally because the organisms find value for themselves in interacting with one another. It is clear that every organism will attempt to maximise its benefit from, and minimise its contribution to, the ecosystem. The ecosystem provides many opportunities for the organism to increase its elegance by leveraging the contributions of others, using the strategies discussed earlier. In fact, these same methods also support increasing elegance through 'cheating' strategies such as parasitism or freeloading. This theoretical perspective is supported by observation: some researchers estimate that 30-50% of all species are parasitic at some point in their lifecycle (Poulin and Morand 2014). It reinforces the point that there is no communal organisation in a natural ecosystem, nor a single commander enforcing community-focussed rules. The levers of regulation for an ecosystem seem quite unlike those of an engineered system of systems.

If we look at ecosystem function from the perspective of the organisms self-organising into a community, we can now begin to see its purpose. The function of the ecosystem is to create or amplify opportunities for its members to improve their elegance and to prevent or mitigate risks to their elegance. It achieves this through the individual actions of its members in the categories listed above, namely behavioural, environmental, and structural adaptations. The result is a whole system which, like all systems, has emergent properties reflecting how efficiently and reliably it executes that function.

As an example of how this works, many animals have a direct personal risk due to the fragility of their closely coupled body architecture, which has many single points of failure. In engineering, redundancy is introduced into closely coupled systems to mitigate such fragility. Most animals do have duplication of some organs, but not others, suggesting that evolution has not found it cost-effective to duplicate these. Instead, the population, which consists of functionally similar individuals, serves to introduce redundancy in support of the species' endurance at an ecosystem level.

Likewise, we have seen how risks naturally arise within the loosely coupled ecosystem in the form of friction in interactions. This can easily arise due to a change in the environment that reduces an organism's efficiency or durability. Functional redundancy at ecosystem level can limit the effects of such friction. For example, in microbial communities, a change in temperature can influence the viability of certain metabolic paths. Any species that can produce the same output via a newly viable path will step in to take the place of a disadvantaged species. Experiments have shown how the mix of species in a community shifts under environmental change, while preserving the community as a whole. The kefir community has been used by humans for at least 5,000 years and produces recognisable kefir despite the secondary flavours being highly sensitive to environmental parameters. The changed flavour profile illustrates that different metabolic processes are at work, despite producing a similar outcome.

4.5 Common design architectures in elegant ecosystems

The above examples illustrate how functional redundancy at ecosystem level can support its enduring viability and resilience as a system of systems. When we look at enduring ecosystems with this lens, we see several recurring system architectures that could plausibly support the elegance of those systems.

At ecosystem level, a drive for efficiency ensures that every underutilised resource will be used. Every creature, no matter how small, has a microbiome. Our own gut microbiome consumes resources that we eat but cannot digest. They convert fibre into valuable nutrients that we can use, in exchange for a regular food supply.

In a previous section, we briefly discussed the incentives for an organism to specialise. Specialists have fewer functions, which means they have fewer resource requirements and a narrower operational range for selection and adaptation. This means they are more likely to be efficient and effective at the things they do. Ecosystems provide the opportunity for many species to specialise, relying on others to fill in missing functions. Non-mobile anemones recruit mobile clown fish to bring them food, while the kefir microbes that cannot cleave milk proteins into more manageable peptides rely on others that can. In this way, we see division of labour emerge at ecosystem level.

That said, in a changing environment, too much specialisation can introduce a significant level of risk to an individual and therefore to any partners who depend on it. Microbial communities such as kefir incorporate a high degree of specialisation, but the way in which it done mitigates that risk.

Kefir contains an enormous variety of species that collectively execute a huge number of metabolic processes, with an even larger number of potential interactions between species. However, it is selective about the species that are allowed into the community; most interlopers, including milk pathogens, are eliminated. Some members are extreme specialists, but their specialist skill is so valuable to the community that it seems almost tailored to their needs. Others are generalists fully capable of surviving in milk on their own. These latter ones are important for functions like cleaving milk proteins, kickstarting fermentation by opening up niches for others. Kefir's specialists tend to be versatile in their specialisms, so can switch their metabolic activity if resource availability or environmental conditions make it preferable. In addition, there is a level of redundancy in specialisms at ecosystem level, with numerous species able to perform them. No two species perform exactly the same set of functions. This means that interchangeable functions can be performed by a range of individuals, each with different niche and environmental tolerances. This provides functional redundancy across a broad niche corresponding to the combined niches of all species that can execute that function. The ecosystem thus ensures a reliable supply of quality resources to its members, under a wide range of conditions.

All these characteristics combine to make kefir, along with other microbial communities, amongst the most enduringly viable ecosystems on Earth. In many ways, their architecture echoes that of tropical rainforests, which similarly use high diversity to make good use of a relatively meagre soil substrate. Division of labour is also found in populations, such as ant and bee colonies, where individuals have roles dependent on their age and expendability rather than through species differentiation.

Another common design architecture in nature leads to a version of quality control. We mentioned earlier that a tiger minimises energy expenditure while hunting by going after the weak, sick, or inexperienced. This removes the individuals that are least effective or efficient at performing their functions and so strengthens the population and ecosystem. It mirrors the process of apoptosis, programmed cell death, which rids the body of weak or damaged cells.

These architectures are found in many durable ecosystems. The mechanisms we have outlined are plausible sources of ilities at ecosystem level. We therefore suggest that these architectures provide the potential for elegance. If, additionally, ecosystem members act to reduce friction within the system, the ecosystem will become more elegant. In the next subsection, we discuss examples of how natural systems mitigate friction.

4.6 How ecosystems tend towards increasing elegance

From the perspective of any individual organism, each other organism in its ecosystem falls into one or more of four categories: predator, competitor, mutualist, and facilitator. Predators introduce risk of injury or to survival. Competitors introduce risk into its access to resources of suitable quality. Mutualists and facilitators provide opportunities to improve its efficiency and reliability, while introducing risk if an existing relationship is disrupted. Friction arises when some network change disrupts an organism's interaction with another such that the other's functional efficiency or reliability is compromised. The effect of this friction on a mutualist or facilitator is much like the effect they would experience from a predator or competitor and will provoke the sort of response a predator or competitor would incite.

The question, therefore, is what an organism can do to avoid turning its allies, or neutral parties, into antagonists, and how it should best respond to competitors and predators impacting their elegance.

What we tend to see in nature is a gradual blurring of or shift between categories. Initially, an organism will adapt to undermine or reduce the impact of a predator or competitor, but over time it will find ways to benefit, as a species, and convert the antagonist into a beneficial partner. We see increasing interdependence between ecosystem partners, which carries an incentive to avoid reducing, or to actively maintain, the suitability of their habitat or availability of their resources. We will sketch a few examples.

Plants are the primary producers in an ecosystem, harvesting the energy of sunlight and storing it in molecules such as glucose. Herbivores are their predators, and plants take a range of measures to repel predators or add risk to foraging. Milkweed, for example, has adapted to produce a toxin that it stores in its leaves. Most herbivores will avoid milkweed as it is foul-tasting as well as poisonous. From the perspective of a herbivore, most plants are facilitators (prey). Poisonous milkweed is neutral, but organisms can become more efficient by expanding their resource networks. The monarch butterfly used this opportunity effectively, with caterpillars developing tolerance for the milkweed toxin. Now, they have a food source most other herbivores cannot eat, which increases their resilience by mitigating risk to their food supply. Moreover, the caterpillars sequester this toxin in their own tissues, rendering both the monarch caterpillar and the butterfly poisonous to predators. Monarch caterpillars have evolved to eat only milkweed, so they are now totally dependent on it. This does increase their vulnerability to a changing climate though, especially with humans encroaching on milkweed habitat.

It is clear that there are many different ways to balance risks and opportunities in solving a problem. With diverse strategic solutions coexisting in the ecosystem, it becomes more resilient as a whole, albeit sometimes at the expense of individual species. Some plants have adapted to predation by leveraging it. Many plants grow better when grazed, with the removal of main tips stimulating the growth of side shoots. Plants recruit their predators to pollinate their flowers and disperse their seeds. In fact Robert Spengler argues that plants such as cereal grasses have become hugely successful by 'seducing humanity into becoming its labour force' (Spengler 2025).

Organisms and their predators are engaged in an arms race of repellents and workarounds. Plants synthesise chemicals to protect themselves against pathogens or parasites. Animals may self-medicate by eating the plants for protection against the same or similar pathogens (Engel 2002). In fact, over time, the impact of a plant chemical on its consumer will often progress from toxin to medicine to essential nutrient. In this way, the plant's actions to increase its own elegance improve the elegance of another, at no significant disadvantage to itself.

Competitors also progress through arms race cycles. Each acts to improve the efficiency of its resource use. In a two-competitor race, one will out-compete the other, but an ecosystem creates relationships and niches that allow both to prevail. A recent experiment tested hundreds of interactions between twelve microbial species and found that in pairs, there was only one survivor, while as a group, the twelve co-exist stably (Chang et al. 2023). Microbes will adapt to confound cheaters who use shared enzymes without contributing any enzymes themselves (Lee et al. 2016). Groundbreaking new research suggests that even plants have evolved means to compete for the attention of a facilitator, by means of structures to remove competing pollen from their pollinators before depositing their own (Minnaar et al. 2019).

Many mutualist relationships originated in facilitation but evolved to create the communication and structure needed for overt partnership. The honeyguide bird actively calls humans to lead them to wild beehives and feeds off the larvae once the honey has been harvested. It is likely that this behaviour originated as a response to an accidental event. Crows and wolves are known to collaborate similarly in joint predation, as do humans and dolphins. When the driver for a mutualism is removed, that mutualism may well break down. Some ants protect acacia trees from herbivores in return for sugar secretions from the trees, but if herbivores are excluded from the area, the trees save energy by producing less sugar and the ants can be outcompeted by a less mutualistic variety.

Communication is an important factor in all natural populations, including microbial ones, for example to coordinate the behavioural changes in social amoebae described earlier. These use quorum sensing, a majority vote style decision making strategy coordinated via tailored signalling chemicals. Effective information gathering and distribution is an important contributor to efficiency and is a key benefit that the ecosystem provides.

It is clear that ecosystem members value the services that the ecosystem provides. Social amoebae that have aggregated into a slug form take samples of their preferred food bacteria along to introduce into the new location (Brock et al. 2011). Recent research has revealed that plant seeds typically house a sample of the original plant microbiome to kickstart a community in their new location. This realisation has stimulated a wide variety of new research, including agricultural applications (Jonkers et al. 2022). An awareness of systems effects is showing potential to transform ecological interventions. For example, reforestation projects are increasingly common, but often centre around tree-planting activities, with mixed success. Recent research has shown, however, that it is far more effective to expand a natural forest by supporting the local frugivores who distribute the tree seeds. This approach results in denser and more diverse tree growth, even at the colonization front (Isla et al. 2024).

In summary, an ecosystem provides the means for its members to increase their elegance beyond what they could achieve as individuals. Where friction arises between members, the ecosystem facilitates actions to mitigate that friction, either by diffusing it, or by developing compensating benefits. Such benefits typically involve both parties improving or maintaining their elegance. It is rarely a zero-sum game.

In a system where all members are working to improve their own elegance and the elegance of their interactions, the overall system will become more elegant. We see that members specialise, which leads to increased diversity and efficiency, which the ecosystem stabilises through redundancy. It is thus plausible that with time, an ecosystem could increase in both diversity and resilience.

It is important to emphasise that this is not a utopian vision. As we mentioned earlier, cheaters will inevitably arise, as cheating is an energy-efficient strategy. The arms race between organisms continues apace. However, overall, natural systems seem to have found ways to trend towards an overall increase in elegance for the majority of parties involved.

5. GENERAL MODEL OF A COMPLEX SOS

Our analysis of the significance of unintended consequences indicated that the risk of system failure arises from friction between the system and allied systems in its operational environment. We picked complex natural ecosystems as a case study to

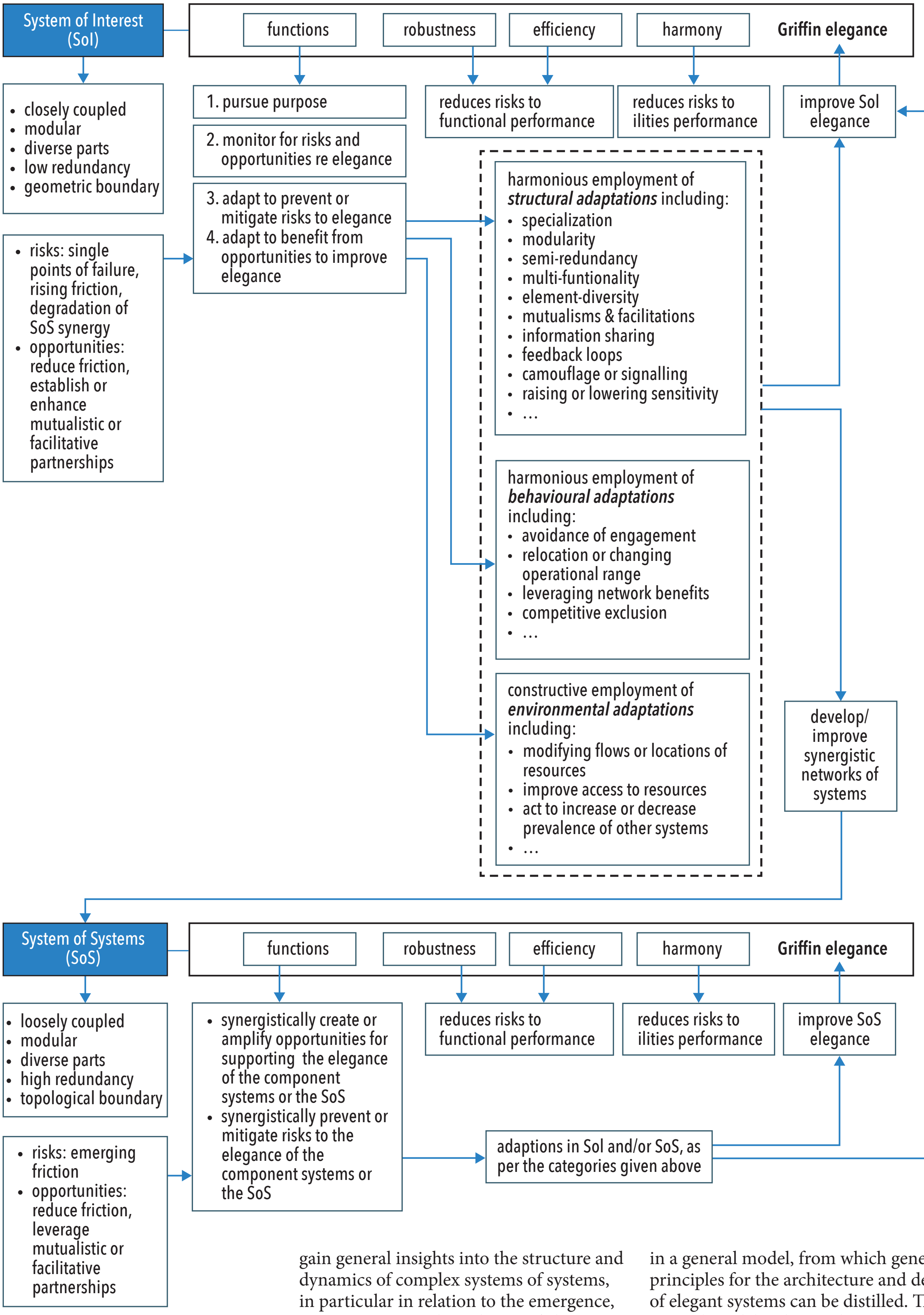


Figure 2. Structure and dynamics for sustaining elegance in SoIs and SoSs

gain general insights into the structure and dynamics of complex systems of systems, in particular in relation to the emergence, improvement, or degradation of elegance at both the system of interest (SoI) level and the system-of-systems (SoS) level.

We can now summarize these insights in a general model, from which general principles for the architecture and design of elegant systems can be distilled. This general model is depicted in Figure 2 and described below.

5.1 General characteristics of a complex SoS and its parts

The general scenario we are trying to model is that of an SoI (the system of interest to be designed or studied) operating in an interacting network of other systems which we will call the SoS (system of systems). To simplify the model, we will cast the SoI as a *closely coupled* complex system (such an organism or an aircraft) and treat it as one of the components of a *loosely coupled* complex SoS (such as a community or a battlegroup). Closely coupled and loosely coupled systems are the polar types of a spectrum of architectures and designs we find in nature and in engineering, but they will suffice to illustrate the key concepts and principles involved across this spectrum.

For reasons already well understood, a closely coupled complex system typically has a modular structure (Simon 1962), a high diversity of closely coupled parts, a geometric boundary that can be defined using entropy gradients, and relatively low redundancy. The latter entails some fragility due to single points of failure, but this is a trade-off because high redundancy in such systems compromises cost-effectiveness. Closely coupled systems have emergent properties, that is, abilities to do things the parts cannot do by themselves (Sillitto et al. 2018). Amongst their emergent properties is their Griffin elegance, a measure of the balance between a system's functional effectiveness, robustness, efficiency, and harmony. As explained earlier, harmony has two components, 'internal harmony' which is intrinsic to the SoI and 'external harmony,' which is related to how well the SoI is adapted to its context. Emergence at the level of the whole has a counterpart at the level of the parts, called 'submergence.' By this is meant that the properties of the parts are to some degree diminished by constraints placed on them by their systemic context (Rousseau 2017b, 2017a). For example, wheels attached to axels attached to carts can roll only in the direction of the cart's momentum, thus contributing to the cart's emergent property of being steerable and movable as a whole.

Complex SoSs typically also have a modular structure and have a high diversity of loosely coupled parts. However SoSs typically have high levels of redundancy, and the loose coupling of the components produces a complex boundary that can be defined using point set topology (Bunge 1992). High redundancy makes SoSs highly robust. Complex SoSs also have emergent properties, including that of Griffin elegance.

In loosely coupled systems, parts can be lost without greatly degrading the properties of the SoS, and parts can leave the SoS and function autonomously for some time, or join another SoS. By the same token, additional parts can be added to the SoS and enhance its properties.

5.2 A new perspective on emergence in complex SoSs

Apart from general characteristics as above, there are factors under which SoIs and SoSs could be understood quite differently in terms of how they function and how they inter-operate. Our study of complex natural ecosystems has revealed a new perspective on the nature of emergence that enriches traditional perspectives on how emergent properties arise.

An ecosystem-type SoS has no objective designer that guides the development and manifestation of the ecosystem functions. Everything the ecosystem achieves is due to the actions of individual parts (component systems), which act and interact in terms of their own self-interest. The parts, if they are organisms, can have purposes, can sense and communicate, and can modulate their behaviours relative to others in order to further their individual goals. The parts have energy, resources, and powers, and these are deployed to effect all the changes that we ascribe to the functioning of the ecosystem. In this sense, the ecosystem as such does not do any work, the parts do everything. However, it is clear that the parts can gain great benefits from their membership of the network, for example via relationships that enable them to:

- reduce the cost of finding resources and removing waste,
- ensure reliable availability and adequacy of resources,
- mitigate risks such as those posed by predators and competitors, and
- exploit synergies to become more specialised and hence more efficient and robust.

These benefits are empowering for the individuals, in that they enable them to do more with less, do things better, faster, and more reliably, and reduce risks of harm or competition. Clearly, the individuals are thus empowered to be able to do or attain things they could not have done on their own. However, the implication of that is that the individuals' enhanced capability is a form of emergence that turns the normal perspective on its head: here, the parts gain emergent properties derived from interactions with their external context. This is emergence at the level of the parts, rather than the classical view of emergence at the level of the whole. It is still the case that the parts, being closely coupled systems, have emergent properties due to the organization of their internal components, but we now see that the parts of the SoS also gain additional emergent powers due to their direct efforts to co-ordinate their actions relative to each other. There is also a new kind of submergence involved here, in that the parts also constrain their behaviour to gain similar benefits for themselves. All this is of course tantamount to saying that within the network the component systems can find support for improving their elegance individually, and this adds up to improving the elegance of the ecosystem. However, we have now realised that the ways in which SoIs and SoSs improve their elegance hinges on complex emergence mechanisms not hitherto fully appreciated.

Of course, our real interest is in finding insights to help engineers attain elegant systems, and for this we have already argued that this requires use of insights about the interactions between the SoI and its associated SoS. However, in the case of engineered systems the SoS can be designed to fulfil an overarching purpose, so the situation is not quite as self-organizing as natural SoSs are. That said, there are many close parallels, and it becomes almost exact if in the natural SoS we have a purposeful component that can establish multiple strong relationships and so act as a kind of 'commander' or 'conductor' of the behaviours of the other parts.

Consider for a simple example a complex SoS such as a football team. It is common to speak of the team as having a purpose (win the trophy) and having emergent powers such as the ability to win football matches in virtue of how its parts are organized. However, on closer inspection it is evident that this is to some degree a convenient gloss on what is actually going on. Teams do not actually do things; the members execute all the actions. Teams don't score goals, individual players do. Teams do not win matches; a sufficient accumulation of individually scored goals do. In the football team, the players gain from co-ordinating their efforts with those of other players, providing benefits (e.g., blocking attackers) and setting up opportunities (passing the ball). In this way the individual players are enabled to achieve outcomes they could not attain by themselves. Thus, the players gain emergent properties provided by their context, the opposite of emergence at the level of the whole, as we discussed above in relation to ecosystems. If we consider the team to include a coach/manager, then we can see the SoS as having a objective purpose but we can still understand all the functions and effects being wholly due to the modulated actions of the individual component systems. The SoS has no objective new powers beyond what can be provided by the components purposefully employing their powers, energies, and resources in ways that co-ordinate between themselves in accordance with their individual values and interests.

This perspective on emergence could potentially resolve several concerns that have historically arisen in connection with the concept of emergence as well as its implications in practical applications.

In terms of the concept, the term 'emergence' technically just refers to a phenomenon and has no explanatory value. Indeed, by not referring to a mechanism it carries an aura of mystery about how the effect comes about. There is now a substantial literature on the apparent mystery of emergence (Axelsson 2022; Bedau and Humphreys 2008; Bunge 2003; Corradini and O'Connor 2013; Gillett 2002, 2006, 2016; Haugen et al. 2023), with no resolution in sight. In application areas, systems scientists and systems engineers often understand the term in opposing ways. To systems scientists, it refers to the things the system can do that the parts cannot do by themselves. To some systems engineers, it refers to behaviours that the system was not designed to have. In the science case emergence is normal and expected, in the engineering case it is unexpected and problematic.

However, the insight that SoI-SoS dynamics involve two kinds of emergence and hence two kinds of submergence cast a new light on the nature of emergence and submergence, one that might resolve these tensions, without deprecating the uses described above. It reveals that the issue at hand is really about how kinds of systemic organization, from complex to complicated, *enable* the SoI. From the perspective of the SoI as a closely coupled system embedded in a loosely coupled SoS, the SoI's capability and potential is enabled by the conjunction of the dynamics of its internal organization and the external orchestration of the allied systems in its environment. In both cases, the enablement that is possible for a given systemic composition is maximised when the friction is minimised.

So, we can now see that concinnity (internal harmony) facilitates the maximum enablement the SoI can gain from its internal organization, and syntony (external harmony) facilitates the maximum enablement the SoI can get from the orchestration of the SoS. Note that by thinking about emergence in terms of enablement we are shifting the focus from phenomena to *mechanisms*. By recognizing that enablement is a balance between mechanisms mediated by *internal organization* and those by *external orchestration*, we open a pathway to tracing and understanding the sources and mechanisms of enablement. In this way we could sidestep the mysteries about emergence that typically arise when aspects of these alternative contributions are missed and/or mixed up.

The counterpart of emergence, namely *submergence*, can now be re-interpreted in the same way. The relevant concept here is "constraint," and once again it is of two kinds. For the SoI, its internal organization constrains the behaviour of its parts, and the orchestration of the SoS constrains the behaviour of the SoI (as a part in the SoS).

Overall, from this shift in focus we can now begin to understand system behaviour and capability as mediated by a balancing act between the enablements and constraints arising from organization and orchestration respectively.

5.3 Dealing with friction, risk, and uncertainty

Natural and engineered systems operate in a world characterised by constant change, risk, uncertainty, and ambiguity. A consequence of this is that an approximately optimised but non-adaptive system would not stay optimal for long – as soon it is deployed it would start to change its environment, and other systems would change in response, and the system will increasingly become less optimal for its purpose. The upshot is that if we wish to ensure enduringly elegant solutions to complex problems, then the solution systems cannot be static – they must incorporate means to monitor for risks to their elegance and means to mitigate or eliminate those risks. Some risks cannot be foreseen or might result in events that happen too quickly for preventative adaptions to be made in time. These risks would entail decline, disruption or loss of access to resources such as energy, materials and information, and waste removal. The main way to minimize the effects of such risks is to work continuously to improve:

- robustness (to reduce need for resources to make repairs or do maintenance),
- efficiency (to reduce resources need to achieve functions), and
- harmony (to reduce resources wasted on overcoming avoidable friction).

Beyond improving elegance, it is also important to build up reserves of resources or increase access to resources that cannot be stockpiled. This would be part of the 'resilience' aspect of robustness. Resistance to, or recovery from, disturbance depends on always having access to sufficient resources.

Therefore, if complex solutions are meant to endure then it is important that they continuously monitor for emerging risks, have means to determine a response that would avoid or mitigate that risk, and means to adapt the system to this end. Three types of adaptations can be involved in a suitable solution, namely adaption to the system's behaviour, environment, or its structure. Different adaptations have different costs and implementation timescales, so depending on the opportunities available and the level and urgency of the risk, different adaptations might be selected or prioritised.

The simplest, fastest, and least costly adaptation is behavioural adaptation, for example moving to another location, changing operational zones, or levering network benefits through information sharing or joining forces against a common threat. This type of adaptation can be effective in the short term but is not typically an enduring solution.

The second possibility is adapting the environment, for example by modifying flows or locations of resources, improving access to resources, or leveraging network effects to increase or decrease the prevalence of similar or other kinds of systems in the proximate territory. This option is typically more costly and takes longer to implement, but it can have more enduring effects than mere behavioural adaption.

The third option is changing the structure of the system. It is the costliest and most time consuming to implement but can be the most impactful and long-lasting change to make. This involves changes to the design of the system, manifesting in features such as specialization, modularity, semi-redundancy, multi-functionality, element-diversity, mutualisms and facilitations, information sharing, feedback loops, camouflage or signalling, and raising or lowering sensitivity. These design upgrades can make the system more complex, but this is not a risk to SoI endurance so long as it is done harmoniously, that is, it does not make the system complicated.

In the face of risk and uncertainty it is important to conserve resources, and hence important to consider implementing lower cost adaptations first. However, the longer the intended lifespan of the solution, the less cost-effective it might become – repeated low-cost adaptations might in the end cost more than an initially higher cost adaptation. It is all a matter or risk level, urgency, and severity.

In many cases of responding to risk, SoIs leverage network effects available to them in an SoS. These clearly support the SoIs to enable them to improve their elegance, and to manage, avoid, or mitigate risks they could not deal with efficiently or at all. In the sense that all the participants in the SoS do this, friction is lowered across the network, and this serves to enhance the elegance of the SoS. This creates a beneficial cycle that supports the endurance of the SoIs and the SoS. Nevertheless, even though the system is improving all the time at all levels, it can never reach an optimal state. Unforeseeable risk events will continue to happen, threatening elegance somewhere and triggering adaptation ripples across the network. In addition, new friction will always arise as individual component systems identify opportunities for improving their elegance at the possible expense of another, triggering further waves of adaptation as the network rebalances. This echoes the insight from studying ecosystems that utopia cannot be reached. That said, healthy ecosystems are always advancing towards greater elegance, creating a protopia.

5.4 Implications of the general model of a complex SoS

Our primary conclusion is that it is important to architect and design the SoI and its relationships with the SoS in an integrated manner, taking account of general insights into the structure and dynamics of their interplay as revealed by studies of enduring complex systems such as ecosystems. In the next section, we will distil such engineering principles derived from our nature study and the general model outlined above.

6. GENERAL PRINCIPLES FOR SYSTEMS ENGINEERING ARCHITECTING

In a changing and uncertain world, every agentic system is at risk of failure due to unforeseeable events and newly emerging risks such as friction from unforeseen consequences and the arrival of new antagonists or competitors. However, in complex SoSs the component systems can also find opportunities to avoid or mitigate those risks and adapt not only to restore their former elegance but become more robust against recurrences of those threats. Natural systems demonstrate many principles supporting an adaptive system's ability to sustain and improve its elegance, and hence its resilience, sustainability, and endurability.

It would be useful for systems engineers to employ these principles when architecting complex systems that are intended to be enduring solutions to complex problems. Here is the list our study uncovered:

Principles for promoting solution success, endurance, resilience, and sustainability

1) Recognize that the success of the SoI cannot be ensured independently from consideration of the current and potential future dynamics within the SoS of which it will be a part.
2) Dealing with realised risks draws on limited resources (time, energy, materials) and this delays or degrades the system's operational performance, which draws on the same reserves. Therefore, the prime directive for sustaining endurance is to:
   a) Minimise the resources needed to do anything;
   b) Maximize accessibility to resources, via reserves for short-term use and means to replenish reserves;
   c) Minimise internal friction by making the internal operations of the system as simple as possible, but no simpler than needed to deliver the specified performance;
   d) Minimize external friction by avoiding actions that may seem predatory or competitive to other SoS components, especially the systems the SoI is dependent on and those that depend on it; and
   e) Ensure potential or realised risks are detected early and dealt with expeditiously, to minimise their impacts on performance or reserves.
3) To minimize resource requirements:
   a) Make the system sufficiently complex to solve the problem, but no more complex. Unnecessary complexity (complicatedness) adds avoidable cost to the design, development, build, maintenance, and operation of the system, while insufficient complexity can extend the problem or make it worse, increasing strains on resources elsewhere in the system of systems;
   b) Monitor for and seize all opportunities to improve elegance by identifying adaptations that increase robustness, efficiency, and harmony; and
   c) Seize every opportunity to gain advantages from the support or behaviour of other members in the SoS.
4) To minimise the likelihood of losses or degradations due to realised risks:
   a) Monitor for the emergence of potential or realised risks, and seize every opportunity to eliminate, mitigate, or overcome them by identifying and implementing adaptations that will preserve or restore the system's elegance;
   b) Seek opportunities to make the system more robust against recurrences of realized risks by identifying adaptions that can increase the system's elegance in advance.
5) Implement these principles to ensure systems continuously and gracefully co-evolve their elegance with that of their associated SoS. This increasingly reduces the likelihood of catastrophic failures at the SoI and SoS levels and increasingly reduces the amplitude of transient disturbances.
6) These principles distilled from enduring ecosystems are general principles for system architecting, and so relevant for every engineering project. They supplement the specialised architecting principles that every project derives from initial activities such as requirements gathering and stakeholder analysis.
7) Remember that architecting principles are important because they ensure that the right things are designed. They support the use of design principles, which ensure that the right things are built. Improving the quality and utility of general architecting principles sits at the root of systems engineering success.

7. EXAMPLES FROM OTHER DISCIPLINES

We derived the principles enumerated above from studies of ecological communities, but our premise was that these principles would apply to the endurability, sustainability, and resilience of complex networked systems of all kinds. We will now defend this assertion based on three examples drawn from fields beyond biology.

Our first example comes from the American multinational conglomerate General Electric (GE). Specifically, we draw insights from its former chairman and CEO Jack Welch. Under Welch's leadership GE became the most valuable corporation in the world. He increased the value of the company more than thirty times over, and during his tenure GE turned out more Fortune 500 CEOs than any other company in history.

A few quotes from him (Krames 2005; Welch 2003, 2006) will serve to illustrate some of our key principles:

- "When the external rate of change exceeds the internal rate of change, the end of your business is in sight."
- "An organization's ability to learn, and translate that learning into action rapidly, is the ultimate competitive advantage."
- "A common mission trap for companies: trying to be all things to all people at all times."

Our second example comes from economics. Here we draw on the work of former hedge fund manager Ray Dalio. Under Dalio's guidance his company, Bridgewater Associates, became the largest and most successful hedge fund in history. Dalio has made a deep study of the ways

Table 1. Optimal implementation of the characteristics of a network-centric organization, derived from (Standish et al. 2017, p. 6)

Characteristic	Optimal Implementation
Vision	Clear vision for business and management is deeply embedded in all company processes.
Culture	Test-and-learn process focused on customer lifetime value and highly collaborative across functions.
Market Insight	Formal tracking of customer, technology, competitor, and disruptor. Rapid evaluation and response.
Data Intelligence	Single view of data is accessible in real-time, and intelligence is operationalized. Decisions automated whenever possible.
Innovation	Defined and prolific innovation process with high number of launches, meticulously evaluated over appropriate time period.
Agility	Flexible processes, systems, and infrastructure for efficient change and partnering capabilities. Fast decision-making.

in which the world order changes, and the factors by which a dominant world power is sustained and declines (Dalio 2021, 2022). Today, America is the dominant world power, following on from the British Empire, which in turn replaced the Dutch Empire, and so on back into history. A central lesson from his study is that for a world power to sustain its dominance it must leverage its trade relationships and improve its efficiency through network effects. It can endure if it is in control of its resources and have internal harmony and external peace. Dalio explains that internal harmony is achieved through laws, and external harmony through treaties. If resource control, external harmony, and/or external peace starts to break down, then the world power will start to decline as costs rise, inefficiencies creep in, and debt increases. The tipping point typically comes when the cost of servicing the country's debt exceeds its defence budget. Unless this is addressed, the way will be opened for another country, one with low debt, strong resource control, internal harmony, and external peace, to become the new world power.

Our third example relates to challenges in organisation design. Over the past few decades, retail organisations have had to undergo significant internal transformation to adapt to successive waves of new technologies and associated changes in shopping behaviour: the internet, digital marketing, ecommerce, mobile commerce, social media, advanced analytics, and so on. As a consultant in ecommerce organisation design and operations during this period, one of us (Billingham) was involved in a joint project between Accenture and Salesforce that, in 2017, published a research report charting a then-emerging trend in retail organisations (Standish et al. 2017). Whereas previously, retailers had been racing to embrace particular changes, now change was happening so fast that they felt the need to design their organisations for the general ability to manage unforeseen change well. The research report dubbed this type of organisation a "network-centric organisation" and identified its key characteristics and optimal implementation (see Table 1). It highlighted an "imperative for retailers and brands to look beyond their own four walls and orchestrate an adaptive network of business partners … based upon an open, interconnected, and collaborative ecosystem."

At that time, internet-driven change resulted in a great deal of internal friction, e.g., the bonuses of store staff being eroded by ecommerce sales. Resolving such conflicts became immediately urgent, but roles with the ability to do so did not even exist. External conflicts arose too: whereas earlier organisations might ignore or even exploit their partners, new organisations had to become much more collaborative, to avoid creating friction for their customers. Truly modern organisations such as Nvidia seem to actively support and promote their partners' elegance, thus building elegantly resilient and harmonious networks. In our view, such an approach will characterise the most successful enterprises of today.

8. CONCLUSION

We have shown that by studying complex natural ecosystems we can distil principles that are relevant to the success, endurance, resilience, and sustainability of complex systems operating in environments that form complex systems of systems. These principles are general in the sense of being scale free and composition independent. We have produced a list of such principles, comprising seven main principles and nine sub-principles. We have argued that these can serve as general principles for system architecting, and hence are relevant for every engineering project, and can usefully supplement the specialised architecting principles that every project derives from early-stage activities such as stakeholder analysis. ■

REFERENCES

- Alhadeff-Jones, M. 2008. "Three Generations of Complexity Theories: Nuances and ambiguities." *Educational Philosophy and Theory* 40 (1): 66–82.
- Axelsson, J. 2022. "What Systems Engineers Should Know About Emergence." *INCOSE International Symposium* 32 (1): 1070–1084. https://doi.org/10.1002/iis2.12982.
- Bedau, M. A., and P. Humphreys. 2008. *Emergence: Contemporary Readings in Philosophy and Science*. MIT Press.
- Billingham, J. A. 2024. "Modelling the Role of Microbial Interactions in Kefir Community Dynamics." MSc diss., Royal Holloway, University of London, GB-ENG.
- Blasche, S., Y. Kim, R. A. T. Mars, D. Machado, M. Maansson, E. Kafkia, A. Milanese, G. Zeller, B. Teusink, J. Nielsen, V. Benes, R. Neves, U. Sauer, and K. R. Patil. 2021. "Metabolic cooperation and spatiotemporal niche partitioning in a kefir microbial community." *Nature Microbiology* 6 (2), Article 2. https://doi.org/10.1038/s41564-020-00816-5.
- Brock, D. A., T. E. Douglas, D. C. Queller, and J. E. Strassmann. 2011. "Primitive agriculture in a social amoeba." *Nature* 469 (7330): 393–396. https://doi.org/10.1038/nature09668.

- Bunge, M. 1992. "System Boundary." *International Journal of General Systems* 20 (3): 215–219. https://doi.org/10.1080/03081079208945031.
- Bunge, M. 2003. *Emergence and Convergence: Qualitative Novelty and the Unity of Knowledge*. University of Toronto Press.
- Chang, C.-Y., D. Bajić, J. C. C. Vila, S. Estrela, and A. Sanchez. 2023. "Emergent coexistence in multispecies microbial communities." *Science* 381 (6655): 343–348. https://doi.org/10.1126/science.adg0727.
- Clausewitz, C. von. 2008. *On War.* Princeton University Press.
- Corradini, A., and T. O'Connor. 2013. *Emergence in Science and Philosophy*. Routledge.
- Dalio, R. 2021. *Principles for Dealing with the Changing World Order: Why Nations Succeed or Fail*. Simon & Schuster UK.
- Dalio, R. 2022. *Principles for Navigating Big Debt Crises*. Simon & Schuster UK.
- Efatmaneshnik, M., and M. J. Ryan. 2019. "On the Definitions of Sufficiency and Elegance in Systems Design." *IEEE Systems Journal* 13 (3): 2077–2088. https://doi.org/10.1109/JSYST.2018.2875152.
- Engel, C. 2002. *Wild Health: How Animals Keep Themselves Well and What We Can Learn From Them*. Weidenfeld & Nicolson.
- Estrada, E. 2024. "What is a Complex System, After All?" *Foundations of Science* 29 (4): 1143–1170. https://doi.org/10.1007/s10699-023-09917-w.
- Francois, C., ed. 2004. *International Encyclopedia of Systems and Cybernetics*. Saur Verlag.
- Frosch, R. A. 1969. "A new look at systems engineering." *IEEE Spectrum* 6 (9): 24–28. https://doi.org/10.1109/MSPEC.1969.5213670.
- Frosch, R. A. 1993. "A Classic Look at Systems Engineering." Speech to IEEE Group on Aerospace and Electronic Systems, IEEE International Convention, New York, 26 March 1969. Reprinted. In T. Hoban & W. M. Lawbaugh (Eds.), *Readings in Systems Engineering: Vol. SP-6102* (pp. 1–7). NASA Scientific and Technical Information Program.
- Gillett, C. 2002. "The varieties of emergence: Their purposes, obligations and importance." *Grazer Philosophische Studien* 65 (1): 95–121.
- Gillett, C. 2006. "The Hidden Battles over Emergence." In *The Oxford Handbook of Religion and Science,* edited by P. Clayton and Z. Simpson, 801–818. OUP Oxford. GB-ENG.
- Gillett, C. 2016. *Reduction and Emergence in Science and Philosophy*. Cambridge University Press.
- Gralka, M., R. Szabo, R. Stocker, and O. X. Cordero. 2020. "Trophic Interactions and the Drivers of Microbial Community Assembly." *Current Biology* 30 (19): R1176–R1188. https://doi.org/10.1016/j.cub.2020.08.007.
- Griffin, M. D. 2007. "System engineering and the 'two cultures' of engineering." *IEEE Engineering Management Review* 35 (3): 44–44.
- Griffin, M. D. 2010. "How Do We Fix System Engineering?" *61st International Astronautical Congress,* Prague, CZ, 27 September – 1 October 2010, 27.
- Haugen, R. A., N. Skeie, G. Muller, and E. Syverud. 2023. "Detecting emergence in engineered systems: A literature review and synthesis approach." *Systems Engineering* 26 (4): 463–481. https://doi.org/10.1002/sys.21660.
- Heylighen, F., P. Cilliers, and C. Gershenson. 2006. "Complexity and Philosophy." In *Complexity, science and society*, edited by J. Bogg and E. Geyer, 117–134. Radcliffe Publishing.
- Hunt, R., and R. L. Colasanti. 2021. "Real communities of virtual plants explain biodiversity on just three assumptions." *In Silico Plants* 3 (1): diab015. https://doi.org/10.1093/insilicoplants/diab015.
- Iandoli, L., L. Piantedosi, A. Salado, and G. Zollo. 2018. "Elegance as Complexity Reduction in Systems *Design." Complexity,* New York, US-NY: 1–10. https://doi.org/10.1155/2018/5987249.
- INCOSE Complex Systems Working Group. 2015. *A Complexity Primer for Systems Engineers* (p. 17). International Council on Systems Engineering (INCOSE).
- Isla, J., M. Jácome-Flores, C. Rigueiro, J. M. Arroyo, P. Jordano, and C. García. 2024. "Animal-mediated seed dispersal and the demo-genetic configuration across plant colonization gradients." *Journal of Ecology* 112 (5): 1013–1025. https://doi.org/10.1111/1365-2745.14280.
- Jenkins, S. 2017. "What, if anything, is systems engineering?" Keynote presentation to the NSF Workshop: Future Directions in Engineering Design and Systems Engineering.
- Jonkers, W., P. E. Gundel, S. K. Verma, and J. F. White. 2022. "Editorial: Seed Microbiome Research." *Frontiers in Microbiology*, 13. https://doi.org/10.3389/fmicb.2022.943329.
- Krames, J. A. 2005. *Jack Welch and The 4 E's of Leadership: How to Put GE's Leadership Formula to Work in Your Organizaion*. McGraw Hill.
- Lee, W., M. van Baalen, and V. A. A. Jansen. 2016. "Siderophore production and the evolution of investment in a public good: An adaptive dynamics approach to kin selection." *Journal of Theoretical Biology* 388: 61–71. https://doi.org/10.1016/j.jtbi.2015.09.038.
- Madni, A. M. 2012. "Elegant systems design: Creative fusion of simplicity and power." *Systems Engineering* 15 (3): 347–354. https://doi.org/10.1002/sys.21209.
- Madni, A. M. 2018. "Design Elegance and Systems Engineering." In *Transdisciplinary Systems Engineering*, 119–132. Springer.
- Minnaar, C., B. Anderson, M. L. De Jager, and J. D. Karron. 2019. "Plant–pollinator interactions along the pathway to paternity." *Annals of Botany* 123 (2): 225–245. https://doi.org/10.1093/aob/mcy167.
- Mittelbach, G. G., and B. J. McGill. 2019. *Community Ecology* (2nd edition). OUP Oxford, GB-ENG.
- Pennotti, M., P. Brook, and D. Rousseau. 2024a. "Systems Engineering and the Pursuit of Elegance: A Transdisciplinary Approach to Complex Problems." *INSIGHT* 27 (2): 8–14. https://doi.org/10.1002/inst.12481.
- Pennotti, M., P. Brook, and D. Rousseau. 2024b. "The Evolution of Systems Engineering as a Transdiscipline." *Systems Engineering* 27 (5): 1–12. https://doi.org/10.1002/sys.21757.
- Pennotti, M., D. Rousseau, P. Brook, and J. Calvo-Amodio. 2025. "On the Meaning, Purpose and Value of Systems Engineering." *INSIGHT* 28 (4).
- Poulin, R., and S. Morand. 2014. *Parasite Biodiversity*. Smithsonian Institution.
- Rousseau, D. 2017a. "Strategies for Discovering Scientific Systems Principles." *Systems Research and Behavioral Science* 34 (5): 527–536.
- Rousseau, D. 2017b. "Three General Systems Principles and their Derivation: Insights from the Philosophy of Science Applied to Systems Concepts." *Proceedings of the 15th Annual Conference on Systems Engineering Research – Disciplinary Convergence: Implications for Systems Engineering Research*, 23-25 March.

- Rousseau, D., J. Billingham, and J. Calvo-Amodio. 2019. "Systemic Virtues as a Foundation for a General Theory of Design Elegance." *Systems Research and Behavioral Science* 36 (5): 656–667.
- Rousseau, D., M. Pennotti, and P. Brook. 2025. "Spectrum and Evolution of Systems Engineering's Guiding Propositions." *Systems Engineering* n/a (n/a). https://doi.org/10.1002/sys.21809.
- Rube Goldberg. 2025. In Wikipedia. https://en.wikipedia.org/w/index.php?title=Rube_Goldberg&oldid=1294450686.
- Salado, A., and R. Nilchiani. 2013. "Using Maslow's hierarchy of needs to define elegance in system architecture." *Procedia Computer Science* 16: 927–936. https://doi.org/10.1016/j.procs.2013.01.097.
- Sillitto, H., J. Martin, D. McKinney, R. M. Griego, D. Dori, D. Krob, P. Godfrey, E. Arnold, and S. Jackson. 2018. *Systems Engineering and System Definitions v1.0*. INCOSE.
- Simon, H. A. 1962. "The Architecture of Complexity." *Proceedings of the American Philosophical Society* 106 (6): 467–482.
- Spengler, R. N. I. 2025. *Nature's Greatest Success: How Plants Evolved to Exploit Humanity*. University of California Press.
- Standish, J., R. Garf, J. Allen, R. Okolie, J. Masket, and J. Rousseau. 2017. *Retail Organizations: The Next Stage of Transformation* (p. 19) [Market Research]. Accenture & Salesforce. https://www.salesforce.com/form/commerce/conf/accenture-retail-transformation-report/.
- Watson, M., and M. Griffin. 2014. "System Engineering: The Discipline of Engineering Elegance. American Institute of Aeronautics and Astronautics (AIAA) Complex Aerospace Systems Exchange (CASE), San Diego, US-CA. https://www.nasa.gov/sites/default/files/atoms/files/system_engineering_discipline_22_nov_2013.docx.
- Watson, M., M. Griffin, P. A. Farrington, L. Burns, W. Colley, P. Collopy, J. Doty, S. B. Johnson, R. Malak, and J. Shelton. 2014. "Building a Path to Elegant Design." *Proceedings of the International Annual Conference of the American Society for Engineering Management*, 1.
- Watson, M., B. Mesmer, and P. Farrington. 2019. "Engineering Elegant Systems: Postulates, Principles, and Hypotheses of Systems Engineering." In *Systems Engineering in Context*, edited by S. Adams, P. A. Beling, J. H. Lambert, W. T. Scherer, and C. H. Fleming, 495–513. Springer International Publishing. https://doi.org/10.1007/978-3-030-00114-8_40.
- Watson, M., B. Mesmer, B., and P. Farrington. 2020. *Engineering Elegant Systems: Theory of Systems Engineering* (Vol. 20205003644).
- Welch, J. 2003. *Jack: Straight from the gut*. Headline.
- Welch, J. 2006. *Winning: The Ultimate Business How-To Book*. HarperCollins Publishers Ltd.

ABOUT THE AUTHORS

David Rousseau is the research director of the Centre for Systems Philosophy in the UK. He is also an INCOSE fellow, a Fellow of the Royal Society of Arts, Commerce and Manufactures, and an associate research professor in the School of Mechanical, Industrial, and Manufacturing Engineering, Oregon State University. David has a BEng from the University of Pretoria, and a PhD from the University of Wales TSD. Early on he served as a lecturer in electronic engineering at the University of Pretoria, then as a project officer with Air Space Control Command in the South African Air Force, and then as manager for Air Force surface-to-air missile systems (SAMS) programmes in the Armaments Corporation of South Africa (ARMSCOR). After moving to the UK, he joined Applied Materials (UK) Ltd as director, eBusiness Solutions. Since 2012 he has been research director of the Centre for Systems Philosophy in the UK. He has published more than 50 articles in peer-reviewed journals and books and discovered several scientific general systems principles. In the International Society for the Systems Sciences (ISSS) he is a past president, an honorary life member, and a past chair of their Board of Trustees. He is a full member of the Omega Alpha Association (the international systems engineering honor society).

Julie Billingham (Rousseau) is the director for strategy at the Centre for Systems Philosophy and a research associate in the Department of Biological Sciences at Royal Holloway, University of London. Her 25-year consulting career in operational research and decision sciences ranged across application areas as diverse as fisheries management, aerospace engineering, and business operations. Prior to joining the Centre for Systems Philosophy, she was a director in transformation services at Salesforce Inc and industry principal at Demandware Inc. She holds a BSc (Hons) in mathematics from the University of Pretoria and an MSc by research from Royal Holloway, University of London. She has published 15 peer-reviewed papers on subjects including resilience engineering, applications of control theory, and natural resource management. She co-authored the leading text on the foundations of General Systems Theory: General Systemology: Transdisciplinarity for Discovery, Insight and Innovation (Springer 2008, reissued in paperback in 2018). She has written more than 10 other published works, particularly in business analytics.

Elegant Solutions to Complex Problems – Case Studies and Examples

Joshua Sutherland, joshua@joshuasutherland.com; **Alfonso Lanza**; **Francesco Dazzi**; **Ken Cureton**; and **Chandru Mirchandani**

■ **ABSTRACT**
A range of experts present a series of short case studies and field examples where elegant solutions emerged to solve complex engineering challenges in network protocols, urban transport, and astrophysics. Their responses illustrate the importance of clarity of purpose and good architecture transforming intricate problem spaces into robust, enduring systems demonstrating that elegance is a core engineering principle, not an afterthought.

INTRODUCTION – JOSHUA SUTHERLAND AND ALFONSO LANZA

Elegance, as defined in NASA's Theory of Elegant Systems (Watson 2020), refers to solutions that are effective, sustainable, and graceful in architecture. It is not mere simplicity, but the result of addressing essential complexity through coherence, clarity, and well-structured design decisions.

For this we asked a range of experts to present a series of short case studies and field examples where elegant solutions emerged to solve complex engineering challenges. You will now read from a diverse set including network protocols, urban transport, and astrophysics. Each story illustrates how clarity of purpose and good architecture can transform intricate problem spaces into robust, enduring systems. Together, they demonstrate that elegance is not an aesthetic afterthought, but a core engineering principle, one that helps us navigate complexity with precision and discipline.

MULTI-MESSENGER ASTRONOMY – FRANCESCO DAZZI

The understanding of the Universe is limited by intricated mysteries that modern astronomy is trying to untangle. The enigmatic nature of dark matter and dark energy, the cataclysmic collisions of black holes and neutron stars, and the origin and role of relativistic cosmic particles are just few examples of interrelated themes that are still not sufficiently explained.

Scientists have turned to an integrative and holistic approach, called "multi-messenger astronomy," to unlock these secrets. Information from different cosmic messengers, such as light, gravitational waves, neutrinos, and cosmic rays, is combined to get insights into the most complex astronomical events. By analysing these different signals together (holistically), scientists can gain a more comprehensive understanding of the sources and processes that produce them.

The observation of a binary neutron star merger, GW170817, in 2017, is one of the most important proofs of the utility and potentiality of multi-messenger astronomy. An event was detected in both gravitational waves and electromagnetic radiation across the entire spectrum, from gamma rays to radio waves. This provided a wealth of information, including a precise localization of the event, confirmation that such mergers are a significant source of heavy elements, and an independent measurement of the Hubble constant that can contributes to the study the nature of dark energy when compared with other similar measurements.

Multi-messenger astronomy is not just a new observational technique, whereas it is an effective and efficient (i.e., elegant) solution to study the Universe complexity with different eyes and from different viewpoints. This real case confirms that complexity demands the application of a transdisciplinary approach where novel knowledge is created by integrating different perspectives ingeniously and transcending the limitations of individual disciplines.

ETHERNET CSMA/CD: AN ELEGANT NETWORK SOLUTION – KEN CURETON

Early computer networks used complicated approaches like central controllers and token-passing systems. Ethernet's CSMA/CD (carrier sense multiple access with collision detection) offered something elegant— much simpler and more effective.

Here's how it works: devices on an Ethernet network listen before they talk. If the line is quiet, they transmit. If two or more devices start talking at once, they detect the collision and back off for a random amount of time before trying again.

This random wait prevents devices from colliding repeatedly. Depending on network traffic, collision rates usually stay under 10%, which means hundreds of different devices can share the same network without any central management telling them when to communicate.

What makes CSMA/CD brilliant is its simplicity. No complex coordination, no handshaking protocols, just devices making their own decisions about when to transmit. This distributed approach eliminated the need for sophisticated network controllers while keeping communication reliable across all kinds of devices on the same Ethernet segment.

CLOUD-BASED SERVICES — CHANDRU MIRCHANDANI

Ensuring resiliency, efficiency, and security in cloud-based systems for large, complex command and control infrastructures—traditionally built as distributed systems—is highly challenging due to intricate interactions and human involvement.

Mirchandani proposed a product-oriented ground data system development model that emphasizes reuse across mission operations. Whereby, leveraging modern processing modules and cloud-based architectures can significantly reduce costs while enhancing reliability and flexibility in space telemetry and command systems.

However, increasing data complexity and diverse processing needs raise the risk of software failures. To address this, it is imperative to identify the interdependencies of data and processes, ensure that appropriate priorities and failure probabilities likelihood are assigned to the interdependent processes and elements which are crucial for optimal testing strategies.

Cloud computing, through virtualization and service-oriented architectures, offers elegant solution of scalable and flexible platforms capable of handling vast data processing demands. These architectures, whether implemented independently or within stakeholder organizations, benefit from mathematical models that link system performance to network quality, enabling robust and efficient data services in evolving control and command environments.

UNDERGROUND RAILWAYS — JOSHUA SUTHERLAND

Moving large numbers of people efficiently through dense cities is a classic complex problem. It involves fluctuating demand, spatial constraints, legacy infrastructure, safety, real-time coordination, and unpredictable disruptions. All interacting within dynamic urban ecosystems.

Many approaches exist above ground, but one elegant solution is the underground railway. Systems like the London Underground, Tokyo Metro, or New York City Subway manage extraordinary complexity below ground, yet deliver a user experience that is simple, reliable, safe, and efficient. Design elements such as frequent service, standardized platforms, simple pricing structures, and abstracted maps, like Harry Beck's 1933 design, make the system intuitive despite its technical depth.

Underground railways demonstrate how layered design, standardization, and interface clarity can transform a complex challenge into a seamless experience.

CONCLUSIONS

A change in business as usual is needed within the systems engineering community if we are to advance the capability of our heuristics. The change requires a coordinated and formal approach to the application and revision of the scope of application and authority of heuristics. The authors call on INCOSE and the systems engineering community at large to leverage I-SHARE as the platform to enable a collaborative and coordinated effort to evolve our heuristics. ■

REFERENCES

- Watson, M. D., B. L. Mesmer, and P. A. Farrington. 2020. "Engineering Elegant Systems: Theory of Systems Engineering." NASA/TP–20205003644, National Aeronautics and Space Administration.

ABOUT THE AUTHORS

Joshua Sutherland is co-chair of INCOSE's Complex Systems Working Group (CSWG) and co-lead of the Future of Systems Engineering (FuSE) scientific foundations initiative. Joshua is the founder and president of Sutherland Systems Engineering Enterprise (SSEE) Ltd, a consultancy and media group specialising in complex systems and technology development. His team delivers projects for global clients including Ford Motor Company, the European Space Agency, and leading technology firms in Silicon Valley and Japan. Joshua is also a fellow at the University of St. Gallen's Center for Aviation and Space Competence and hosts the world's leading podcast in the systems engineering space.

Alfonso Lanza holds engineering degrees from Imperial College London and the University of London.

Francesco Dazzi is a senior technologist at the Italian National Institute for Astrophysics (INAF), president 2022-2023 of AISE (INCOSE Chapter Italy), and member of the INCOSE Institute for Technical Leadership. He has more than 20 years of experience mainly dedicated to the construction of ground-based telescopes and technical management of scientific projects. He has a degree in physics, a PhD in mathematics and physics, and more than 250 publications including journal articles and conference proceedings. In 2018 he obtained the CSEP (certified systems engineering professional) certification.

Kenneth L. Cureton holds a BS in physics from California State University Los Angeles and an MS in systems architecting and engineering from USC, where he has served as a part-time lecturer since 1996. He retired from Boeing (Rockwell) after 29 years, with additional experience spanning 16 years in commercial and government sectors. His expertise includes enterprise systems architecting, system-of-systems modeling and integration, complex networked systems interoperability, resilience engineering, and avionics design for manned and unmanned space systems. Cureton chairs the INCOSE Resilient Systems Working Group and serves on AIAA Space Settlement and Systems Engineering Technical Committees. He previously co-chaired the IEEE SMC MBSE Working Group and served as technical council chair emeritus for the NCO Industry Consortium.

Dr. Chandru Mirchandani is a fellow of INCOSE, an associate fellow of AIAA, and a senior member of IEEE. He began his career in R&D of integrated circuits at Hewlett-Packard and later supported NASA's Goddard Space Flight Center while at Lockheed Martin, developing advanced VLSI-based telemetry systems and leading projects on intelligent sensor and satellite networks in collaboration with JPL. His work has advanced ASIC and FPGA technologies, XBAND processing, and the infusion of COTS devices into space and ground systems. Currently a Leidos technical fellow, he serves as a qualified system architect and principal systems engineer specializing in reliability, maintainability, and availability (RMA). He has developed innovative methods for fault modeling, software reliability prediction, and system analysis, and contributes to the Leidos Digital Engineering Center of Excellence. Dr. Mirchandani teaches systems engineering and related courses as an adjunct professor at George Washington University. A Fulbright specialist (2012), he has taught internationally and frequently lectures on reliability engineering, risk management, and enterprise systems.

Elegant Solutions to Complex Problems – Recommended Resources

Joshua Sutherland, joshua@joshuasutherland.com; **Alfonso Lanza**; **Peter Brook**; **Alejandro Salado**; and **Chandru Mirchandani**

ABSTRACT
A range of experts provide guidance from foundational theories to emerging perspectives on what materials have helped them understand the field, presenting a curated set of books, papers, frameworks, and conversations that have helped systems engineers deepen their understanding of complexity and refine their ability to recognize—or design—elegant solutions.

INTRODUCTION – JOSHUA SUTHERLAND AND ALFONSO LANZA

Understanding complexity and navigating its vast literature and knowledge base can be characterized as a complex task in its own right!

To help with this, we asked several experts to provide guidance on what materials have helped them understand the field, as such, this article presents a curated set of books, papers, frameworks, and conversations that have helped systems engineers deepen their understanding of complexity and refine their ability to recognize — or design — elegant solutions.

From foundational theories to emerging perspectives, these resources provide practical and conceptual grounding for navigating complexity with discipline, coherence, and purpose.

Is something missing? We would be glad to hear from you: Joshua.Sutherland@incose.net.

PETER BROOK, INCOSE FELLOW

I began to understand complexity when I read *Complexity Theory and the Social Sciences* by Byrne and Callaghan (2013), at the suggestion of Hillary Sillito. The first half is a general treatment of complexity science, albeit academic, and the second half opened my eyes to how complex social issues might be addressed. The latest edition (2023) gives greater insight into the strengths and limitations of modelling and prediction techniques in the social domain— as they apply to COVID, for example.

Michael Pennotti drew my attention to Snowden's Cynefin framework which shows how one can link the complexity of our problems to general heuristics for solving them.

On David Rousseau's recommendation, I am currently reading *Making Sense of Chaos–A Better Economics for a Better World* by J Doyne Farmer, a founder member of the Santa Fe Institute. It centres on economic systems, but also takes in social, ecological and climate systems. (Farmer argues for fine-grain modelling.) An excellent guide to complexity in 21st Century world problems.

These texts are deepening my understanding of how we could improve our delivery of elegant solutions to the broadest range of complex problems through a better understanding of the underlying systems phenomena. Turning this into actionable advice is an enduring challenge for systems engineering.

Along the way, I have enjoyed *Complexity – A Very Short Introduction* by John H Holland (another Santa Fe alumnus) and *Do Dice Play God?* by Ian Stewart, who discusses the mathematics of uncertainty, with excellent sections on modern complexity theory. These are both good entry-level texts for general readers wishing to become more engaged with the subject.

Maybe the most influential text I have come across related to complexity is Simon's foundational *Architecture of Complexity*. It explains many things at the same time: how multi-level systems evolve efficiently; the importance of modularity; and how interactions unfold over time These factors are brought together coherently in what he calls near-decomposable architectures, built around subsystems where interactions are strongest. These nodes can act as gathering points for progressive architectural decomposition and for governance. I learn something new every time I read it.

JOSHUA SUTHERLAND

Three important resources and experiences have shaped how I think about complexity problems, and I believe they'll be helpful to others exploring the same space.

First is the book *Complexity: A Guided Tour* by Melanie Mitchell. It offers a helpful overview and clarified for me just how much of the complexity field lacks consensus. That realisation helped me think more critically and work more flexibly within ambiguity.

Second is being part of the INCOSE Complex Systems Working Group (CSWG). Ongoing conversations about emergence, definitions, and the scientific foundations of complexity have expanded my understanding of what makes a system "complex" in the first place. The mentorship of group members has also taught me a great deal about the intellectual history of the field.

Finally, designing and running experiments with Prof. Olivier de Weck's research group has enabled us to explore how a scientific consensus around complexity in systems engineering might eventually be formed.

Together, these experiences continue to shape how I approach elegance in systems practice, not as simplification, but as clarity within complexity.

ELEGANT SOLUTIONS TO COMPLEX PROBLEMS – IMPORTANT RESOURCES FROM NASA, INCOSE AND MIT – ALFONSO LANZA AND JOSHUA SUTHERLAND

NASA has been at the forefront of exploring how to engineer elegant solutions to complex problems — systems that are not only effective and robust, but also efficient and gracefully integrated. An important set of resources comes from the NASA Systems Engineering Research Consortium, led by Michael D. Watson and colleagues, which developed a theoretical and practical framework for Engineering Elegant Systems.

These resources include foundational publications such as "Engineering Elegant Systems: Theory of Systems Engineering" (Watson et al. 2020a) and "The Practice of Systems Engineering" (Watson et al. 2020b), which articulate a framework grounded in formal postulates, guiding principles, and strategic approaches to managing complexity. One of the key theoretical insights—Hypothesis 2 — asserts that any real system will have complexity equal to or greater than the minimum complexity required to fulfill all intended goals. This "ideal" system complexity is defined not by simplicity in any one dimension, but by achieving a holistic balance across structural, functional, temporal, and contextual dimensions. The implication is clear: overly complex solutions may work, but only the minimally complex system that satisfies all goals can be considered elegant.

This aligns closely with the proposed First Law of Systems Science and Engineering: Conservation of Complexity, proposed by de Weck and colleagues at MIT (de Weck 2023), which formalizes complexity as a quantity traded off against system performance and development effort. Both frameworks emphasize that while complexity cannot be arbitrarily minimized, it can — and should — be optimally distributed to meet the demands of the system. Together, these perspectives reinforce the notion that elegance is not the absence of complexity, but its disciplined orchestration.

Appendix B of Watson's theory document further catalogues key properties of complex systems — aggregation, emergence, nonlinearity, and bounded optimality— all of which must be understood and managed to approach minimal complexity in practice. These insights are further elaborated in the INCOSE Complexity Primer (2015, 2020), which complements NASA's contributions with practical heuristics and evolving systems science principles.

For engineers, researchers, and policymakers, these resources offer not just guidance — but a unified conceptual foundation — for designing elegant, resilient, and mission-aligned systems in the face of irreducible complexity.

ELEGANT SOLUTIONS TO COMPLEX PROBLEMS – TOWARDS A SCIENCE OF ELEGANCE – ALEJANDRO SALADO

My research has explored the intersection of engineering, aesthetics, and complexity theory to develop a rigorous understanding of elegance in systems engineering. Traditional definitions of elegance are often subjective. Instead, we have proposed a structural definition based on Maslow's hierarchy of needs, where elegance is defined as the satisfaction of functional, performance, availability, efficiency, and adaptability needs without major intervention from the user or owner (Salado and Nilchiani 2013a, 2013b). This framework allows for consistent assessment and comparison of candidate designs.

We have extended this approach by conceptualizing elegance as effective complexity, a perceptual balance between excessive order and randomness, shown to correlate strongly with perceived architectural quality (Iandoli, Salado, and Zollo 2018). Drawing from neuroaesthetics and Gestalt psychology, we demonstrated that visual heuristics commonly used in art, such as symmetry, grouping, and contrast, can be applied to reduce complexity and improve stakeholder communication (Salado, Iandoli, and Zollo 2016; Salado and McDermott 2018).

Ultimately, elegance emerges not from process alone but from a triad of enablers, facilitation, and human talent (Salado and Nilchiani 2013b). By integrating the cognitive processes behind aesthetic reasoning, we can design systems that are not only technically sound but also meaningful and resilient (Salado, Iandoli, and Zollo 2019, 2022).

CHANDRU MIRCHANDANI

- Books: Simon French and Roger Clemons books on decision making tools; Book by Camm on data analytics and Modarres book on reliability engineering and risk analysis.
- Papers: Some papers that have helped me in understanding the elegant complexity of cloud-based systems and system of systems architectures:
- Banzai, Takayuki et al; "D-Cloud: Design of a Software Testing Environment for Reliable Distributed Systems Using Cloud Computing Technology," 10th IEEE/ACM International Conference, 2010.
- Chorng-Shiuh Koong et al; "The Architecture of Parallelized Cloud-based Automatic Testing System," Seventh International Conference on Complex, Intelligent, and Software Intensive Systems, 2013.
- Knorr, Eric et al; "What cloud computing really means," InfoWorld, 2008.
- Ahmed Shouman et al; "Service Oriented Architecture for Remote Sensing Satellite Telemetry Data Implemented on Cloud Computing," I. J. Information Technology and Computer Science, 2013.
- Mirchandani, Chandru; "Cloud-Based Ground System for Telemetry Processing," Complex Adaptive System Conference, CA, 2015.
- Discussions: INCOSE Complex Systems Working Group (CSWG). Ongoing conversations about emergence, definitions, and the scientific foundations of complexity involvement in the tradecraft guide and the primer towards what makes a system "complex" in the first place. ■

REFERENCES

- Byrne, D., and G. Callaghan. 2013. *Complexity Theory and the Social Sciences: The State of the Art*. Routledge.
- De Weck, O. 2023. "The First Law of Systems Science: Conservation of Complexity.' INCOSE International Workshop, Torrance, US-CA.
- Farmer, J. D. 2024. *Making Sense of Chaos: A Better Economics for a Better World*. Yale University Press.
- Holland, J. H. 2014. *Complexity: A Very Short Introduction*. Oxford University Press.

- Iandoli, L., A. Salado, and G. Zollo. 2018. "Elegance as Complexity Reduction in Systems Design." *Complexity*.
- INCOSE. 2015. "A Complexity Primer for Systems Engineers." International Council on Systems Engineering.
- INCOSE. 2020. "A Complexity Primer for Systems Engineers." International Council on Systems Engineering.
- Mitchell, M. 2009. *Complexity: A Guided Tour*. Oxford University Press.
- Salado, A., and R. Nilchiani. 2013a. "Using Maslow's Hierarchy of Needs to Define Elegance in System Architecture." Procedia Computer Science.
- Salado, A., and R. Nilchiani. 2013b. "Elegant Space Systems: How Do We Get There?" In 2013 IEEE Aerospace Conference.
- Salado, A., L. Iandoli, and G. Zollo. 2016. "Painting Systems: From Art to Systems Architecting." In INCOSE International Symposium.
- Salado, A., and T. McDermott. 2018. "Art and Architecture: Effectively Communicating Models of Systems." In 2018 Annual IEEE Systems Conference (SYSCON).
- Salado, A., L. Iandoli, and G. Zollo. 2019. "The Role of Aesthetic Reasoning in Knowledge Management: The Case of Elegant Systems Architecture Design." *Knowledge Management Research & Practice*.
- Salado, A., L. Iandoli, and G. Zollo. 2022. "Towards a New Science of Interfaces: Bridging Art and Engineering." In *Thinking*, edited by N. Rezaei and A. Saghazadeh. Integrated Science: Springer.
- Simon, H. A. 1962. "The Architecture of Complexity." Proceedings of the American Philosophical Society 106 (6): 467-482. University of Pennsylvania Press.
- Stewart, I. 2019. *Do Dice Play God?: The Mathematics of Uncertainty* Profile Books Ltd.
- Watson, M. D., B. L. Mesmer, and P. A. Farrington. 2020a. "Engineering Elegant Systems: Theory of Systems Engineering." NASA/TP–20205003644, National Aeronautics and Space Administration.
- Watson, M. D., B. L. Mesmer, and P. A. Farrington. 2020b. "Engineering Elegant Systems: The Practice of Systems Engineering." NASA/TP–20205003646, National Aeronautics and Space Administration.

ABOUT THE AUTHORS

Joshua Sutherland is co-chair of INCOSE's Complex Systems Working Group (CSWG) and co-lead of the Future of Systems Engineering (FuSE) scientific foundations initiative. Joshua is the founder and president of Sutherland Systems Engineering Enterprise (SSEE) Ltd, a consultancy and media group specialising in complex systems and technology development. His team delivers projects for global clients including Ford Motor Company, the European Space Agency, and leading technology firms in Silicon Valley and Japan. Joshua is also a fellow at the University of St. Gallen's Center for Aviation and Space Competence and hosts the world's leading podcast in the systems engineering space.

Alfonso Lanza holds engineering degrees from Imperial College London and the University of London.

Peter Brook is an INCOSE fellow, a fellow of the UK's Royal Academy of Engineering, and is with Dashwood Systems Engineering in the UK. He has consulted widely across the UK government and DARPA in the US. He spent most of his employed career in defence R&D establishments, filling a number of senior posts, including director of systems engineering for DERA, with secondments to MOD: head of the integration authority and director of science (land systems). He has lectured widely, and co-authored a systems engineering textbook. His recent research interests have focused on enterprise systems engineering and systems of systems. Peter has first degrees in physics: BSc Wales (Swansea), MSc London (UCL, External), with an early career spent in researching solid-state microwave systems. He was awarded a visiting professorship in systems engineering at the Cranfield University/Defence Academy and retains an active presence in the academic and professional communities. He served two terms on the INCOSE board of directors (1996-99).

Alejandro Salado is an associate professor of the institute of technology systems engineering and the director of systems engineering with the department of systems engineering at the University of Arizona. He conducts research in multiple areas, trying to unveil the theoretical foundations of systems engineering that can underpin methods that improve practice. Alejandro holds a PhD in systems engineering from the Stevens Institute of Technology.

Dr. Chandru Mirchandani is a fellow of INCOSE, an associate fellow of AIAA, and a senior member of IEEE. He began his career in R&D of integrated circuits at Hewlett-Packard and later supported NASA's Goddard Space Flight Center while at Lockheed Martin, developing advanced VLSI-based telemetry systems and leading projects on intelligent sensor and satellite networks in collaboration with JPL. His work has advanced ASIC and FPGA technologies, XBAND processing, and the infusion of COTS devices into space and ground systems. Currently a Leidos technical fellow, he serves as a qualified system architect and principal systems engineer specializing in reliability, maintainability, and availability (RMA). He has developed innovative methods for fault modeling, software reliability prediction, and system analysis, and contributes to the Leidos Digital Engineering Center of Excellence. Dr. Mirchandani teaches systems engineering and related courses as an adjunct professor at George Washington University. A Fulbright specialist (2012), he has taught internationally and frequently lectures on reliability engineering, risk management, and enterprise systems.

Elegant Solutions to Complex Problems – Perspectives and Practice

Joshua Sutherland, joshua@joshuasutherland.com; **Alfonso Lanza**; **Kristin Giammarco**; **Michael Pennotti**; **Dov Dori**; and **Louise Harney**

ABSTRACT
Experts provide their personal perspectives, conceptual frameworks, and behavioral insights that shape how systems engineers approach complex challenges to illuminate how elegance also emerges from how we think, frame, and engage with complexity.

INTRODUCTION – JOSHUA SUTHERLAND, ALFONSO LANZA

While elegant solutions are often revealed through concrete systems and technical examples, elegance also emerges from how we think, frame, and engage with complexity. This collection brings together personal perspectives, conceptual frameworks, and behavioral insights that shape how systems engineers approach complex challenges.

Each contribution reflects a different angle on what makes a system, not just functional, but elegant. Together, these perspectives remind us that elegance is not a fixed outcome, but a way of thinking, seeing, and practicing systems engineering.

FINDING THE ANGEL IN THE MARBLE – KRISTIN GIAMMARCO

In 1883, George Frederick Pentecost shared a story of a sculptor who looked at a rough stone block and declared, "There is a beautiful angel in that block of marble, and I am going to find it." The sculptor explained his intent to remove the excess stone, with careful chiseling, to reveal a masterpiece. Achieving elegance in complex systems demands that we approach system behavioral design work as sculptors. Emergence engineers will start from a superset of possible behaviors both wanted and unwanted (raw marble). Then they will methodically eliminate the unwanted behaviors, shaping for overall system effects, to unveil an elegant and well-behaved complex system (the angel in its splendor). This approach of "finding" an angel in the marble is one that could be taken by systems engineers to likewise find elegant expressions of complex systems, using Monterey Phoenix to first generate an exhaustive (up to a scope limit) search space and Artificial Intelligence -assisted methods to then help designers find, and decide to keep or reject, instances of behavior within that search space. Let us consider this sculptor's mindset as we pursue the engineering of elegant complex systems.

SYSTEMS ENGINEERING THROUGH A SYSTEMS LENS – MICHAEL PENNOTTI

I have frequently found myself in the position of having to explain systems engineering to technical and business leaders. On those occasions, I found the specialized process language we often use as practitioners unhelpful. To remedy this, my Stevens colleague Bill Robinson and I devised a framework we call the systems lens. Channeling our more than five decades of combined systems engineering experience at Bell Labs, our framework describes what systems engineers do, not how we do it. And it does so in plain English that leaders can relate to. The framework defines four categories of activities:

- Defining what to build and why
- Bringing solutions to life
- Ensuring that systems work and are robust
- Managing evolution and deciding what's next.

We emphasize that these categories are not meant to be sequential. At any point in time, systems engineers might find themselves engaged in one or more of them and they might have to shift from any one to any other, as the situation warrants.

We found the framework very intuitive and helpful in conversations with leaders, and we used it to structure successful workshops and courses for over 20 years. After one such workshop, we received what I regard as our highest compliment as instructors when a workshop participant summarized his experience by saying, "I came here hoping to learn how to cook. But you're trying to make me a chef!" What if that became our goal?

OBJECTS AND PROCESSES AS A MINIMAL UNIVERSAL ONTOLOGY – DOV DORI

Guided by the principle of minimalism, a minimal universal ontology is the smallest set of conceptual building blocks necessary and sufficient to model systems and phenomena in the universe. I am working on proving the object-process theorem, which asserts that objects, processes, and relations among them constitute a minimal universal ontology. The theorem lays down the theoretical foundation for a formal conceptual modeling language capable of specifying any system. object-process methodology (OPM), an international standard ISO 19450, is founded on this theorem. When I invented OPM back in 1995, the object-oriented paradigm, which was used in programming, was gaining popularity also in the modeling, analysis, and design domains. However, I had an intuition that objects alone are only half of the story – they are good at representing the structural aspect of the system. To complete the picture with the procedural-dynamic system aspect, processes are mandatory as first-class citizens beside objects rather than being subordinates of objects as "methods." The first book on OPM, published in 2002 (Dori 2002), is based on this paradigm and presents the visual and textual modalities of OPM as an elegant modeling language and methodology. The second one (Dori 2016) elaborates on ideas and discusses how the various SysML diagrams can be expressed elegantly with this minimal universal ontology.

HANDLING COMPLEXITY — WHAT MATTERS MOST — MINDSET & BEHAVIORS — LOUISE HARNEY

One may find themselves searching for the methodology or technique which is recommended for handling complexity, that will solve your problems. Perhaps it's model-based, perhaps it focusses on communication or soft skills. The reality seems to be quite different — complexity flexes and adapts, complexity has different characteristics, types, and categories. Complexity in essence is somewhat unique in each instance and perhaps even dependent on each observer. The mindset for handling complexity, though, can be constant, supporting us as we get comfortable not knowing the ins and outs, not decomposing everything into its building blocks; responding to outputs and signposts, rather than analyzing the individual factors that made it so. Handling complexity requires open, questioning behaviors; seeking feedback and diverse views; taking small actions and testing the results, learning from experience. When ChatGPT was asked to summarize this behavior in one or a few words, one of the options was "systems thinking". Yes, I suppose it is systems thinking, more so than ever before, in each of our behaviors and everything we do. ■

REFERENCES

- Dori, Dov. 2002. *Object-Process Methodology – A Holistic Systems Paradigm*. Berlin, Heidelberg, New York: Springer Verlag.
- Dori, Dov. 2016. *Model-Based Systems Engineering with OPM and SysML*. New York: Springer.

ABOUT THE AUTHORS

Joshua Sutherland is co-chair of INCOSE's Complex Systems Working Group (CSWG) and co-lead of the Future of Systems Engineering (FuSE) scientific foundations initiative. Joshua is the founder and president of Sutherland Systems Engineering Enterprise (SSEE) Ltd, a consultancy and media group specialising in complex systems and technology development. His team delivers projects for global clients including Ford Motor Company, the European Space Agency, and leading technology firms in Silicon Valley and Japan. Joshua is also a fellow at the University of St. Gallen's Center for Aviation and Space Competence and hosts the world's leading podcast in the systems engineering space.

Alfonso Lanza holds engineering degrees from Imperial College London and the University of London.

Kristin Giammarco is the principal investigator for Monterey Phoenix, the U.S. Navy-developed language, approach, and tool for exposing and controlling unexpected emergent behaviors in systems. As a tenured associate professor in the department of systems engineering at the Naval Postgraduate School (NPS), she taught and researched formal methods for system behavior modeling, and managed systems engineering distance learning programs. Prior to NPS she worked as a chief engineer for the U.S. Army, where she developed quick-turn model-based systems engineering analysis techniques with her team. Her early career experience was in testing and risk analysis for communications systems at MITRE, Lockheed Martin, Bellcore, and AlliedSignal Aerospace. Dr. Giammarco is a 20-year member of INCOSE with a PhD in software engineering from NPS, an MS in systems engineering management from NPS, and a BE in electrical engineering from Stevens Institute of Technology.

Michael Pennotti is an INCOSE fellow and was formerly a distinguished service professor in the school of Systems and Enterprises at Stevens Institute of Technology, Hoboken, NJ, USA. He served as associate dean for academics in the school and later, director of systems and software programs. Prior to joining Stevens in 2001, Mike spent 20 years as a systems engineering practitioner and leader at Bell Laboratories, designing and improving the operational performance of four generations of undersea surveillance systems for the U.S. Navy, and another 10 years applying systems principles to enterprise systems in executive positions with AT&T, Lucent Technologies, and Avaya. He is a senior life member of the IEEE and was a co-founder and coach of the INCOSE Technical Leadership Institute. He holds PhD and MS degrees in electrical engineering from the Polytechnic Institute of New York, a BEE from Manhattan College, and is a graduate of the AEA/Stanford Executive Institute for Technology Executives.

Dov Dori is a pioneer and fellow of INCOSE, life fellow of IEEE, and fellow of IAPR and AAIA. He is professor emeritus of systems engineering and head of the enterprise systems modeling laboratory at the faculty of data and decision sciences, Technion, Israel Institute of Technology. He was a visiting professor at MIT intermittently between 1999 and 2020. In 1993, Dr. Dori invented object-process methodology, OPM, which has become ISO 19450 international standard. He wrote two books on OPM, which have been central to model-based systems engineering (MBSE) and provided the basis for the MBSE edX certificate program and MOOC series. Prof. Dori has supervised over 60 graduate students and authored 400 publications, cited over 8600 times. He chaired nine international conferences and was associate editor of *IEEE T-PAMI* and *Systems Engineering*. He was co-founder and co-chair of the IEEE Society of Systems, Man, and Cybernetics technical committee on MBSE. He has received various research, innovation, and teaching awards, including the INCOSE pioneer award and INCOSE propeller hat award.

Louise Harney is a chief systems specialist at Leonardo, focused on system safety and digital engineering for complex systems in future combat air, and radar and advanced targeting. She is leading the development of INCOSE's Tradecraft Guide for Systems Engineers.

Pennotti et al. continued from page 45

David Rousseau is an INCOSE fellow, a fellow of the Royal Society of Arts, Commerce and Manufactures, an associate research professor in the School of Mechanical, Industrial, and Manufacturing Engineering, Oregon State University, and is with the Centre for Systems Philosophy in the UK. David has a BEng from the University of Pretoria, and a PhD from the University of Wales TSD. Early on he served as a lecturer in electronic engineering at the University of Pretoria, then as a project officer with Air Space Control Command in the South African Air Force, and then as manager for Air Force surface-to-air missile systems (SAMS) programmes in the Armaments Corporation of South Africa (ARMSCOR). After moving to the UK, he joined Applied Materials (UK) Ltd as director, eBusiness Solutions. Since 2012 he has been research director of the Centre for Systems Philosophy in the UK. He has published more than 50 articles in peer-reviewed journals and books and discovered several scientific general systems principles. In the International Society for the Systems Sciences (ISSS) he is a past president, an honorary life member, and a past chair of their board of trustees. He is a full member of the Omega Alpha Association (the international systems engineering honor society).

Peter Brook is an INCOSE fellow, a fellow of the UK's Royal Academy of Engineering, and is with Dashwood Systems Engineering in the UK. He has consulted widely across the UK government and DARPA in the US. He spent most of his employed career in defence R&D establishments, filling a number of senior posts, including director of systems engineering for DERA, with secondments to MOD: head of the integration authority and director of science (land systems). He has lectured widely, and co-authored a systems engineering textbook. His recent research interests have focused on enterprise systems engineering and systems of systems. Peter has first degrees in physics: BSc Wales (Swansea), MSc London (UCL, External), with an early career spent in researching solid-state microwave systems. He was awarded a visiting professorship in systems engineering at the Cranfield University/Defence Academy and retains an active presence in the academic and professional communities. He served two terms on the INCOSE board of directors (1996-99).

Javier Calvo-Amodio is chair of the INCOSE Systems Science Working Group, a fellow of the American Society for Engineering Management, an associate professor of nuclear engineering and industrial technology at the University of Idaho, a guest professor at Keio University in Japan, a visiting fellow at the Centre for Systems Studies at Hull University, UK, and deputy editor of the Systems Research and Behavioral Sciences Journal. At the University of Idaho, his research focuses on developing a fundamental understanding of the scientific foundations of systems as they apply to industrial engineering, systems engineering, and engineering management to enable better engineering of organizations.

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

Call for Papers

The *Systems Engineering* journal 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 activities involve the technologies and system management approaches needed for

- definition of systems, including identification of user requirements and technological specifications;
- development of systems, including conceptual architectures, tradeoff of design concepts, configuration management during system development, integration of new systems with legacy systems, integrated product and process development; and
- deployment of systems, including operational test and evaluation, maintenance over an extended life-cycle, and re-engineering.

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 supports these goals by providing a continuing, respected publication of peer-reviewed results from research and development in the area of systems engineering. Systems engineering is defined broadly in this context as an interdisciplinary approach and means to enable the realization of successful systems that are of high quality, cost-effective, and trustworthy in meeting customer requirements.

The *Systems Engineering* journal is dedicated to all aspects of the engineering of systems: technical, management, economic, and social. It focuses on the life-cycle processes needed to create trustworthy and high-quality systems. It will also emphasize the systems management efforts needed to define, develop, and deploy trustworthy and high quality processes for the production of systems. Within this, *Systems Engineering* is especially concerned with evaluation of the efficiency and effectiveness of systems management, technical direction, and integration of systems. *Systems Engineering* is also very concerned with the engineering of systems that support sustainable development. Modern systems, including both products and services, are often very knowledge-intensive, and are found in both the public and private sectors. The journal emphasizes strategic and program management of these, and the information and knowledge base for knowledge principles, knowledge practices, and knowledge perspectives for the engineering of systems. Definitive case studies involving systems engineering practice are especially welcome.

The journal is a primary source of information for the systems engineering of products and services that are generally large in scale, scope, and complexity. *Systems Engineering* will be especially concerned with process- or product-line–related efforts needed to produce products that are trustworthy and of high quality, and that are cost effective in meeting user needs. A major component of this is system cost and operational effectiveness determination, and the development of processes that ensure that products are cost effective. This requires the integration of a number of engineering disciplines necessary for the definition, development, and deployment of complex systems. It also requires attention to the lifecycle process used to produce systems, and the integration of systems, including legacy systems, at various architectural levels. In addition, appropriate systems management of information and knowledge across technologies, organizations, and environments is also needed to insure a sustainable world.

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, with the intent to achieve published papers that

- relate to the field of systems engineering;
- represent new, previously unpublished work;
- advance the state of knowledge of the field; and
- conform to a high standard of scholarly presentation.

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:

https://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.

36th Annual INCOSE International Symposium

hybrid event

Yokohama, Japan
13–18 June 2026

INCOSE®

Call for Submissions

Call for Papers

Call for Panels/ Roundtables

Call for Tutorials

Call for paperless presentations

Visit **www.incose.org/symp2026** and contact us **TODAY** – The INCOSE Events Team