



# Envisioning Systems Engineering as a Transdisciplinary Venture

Hillary Sillitto  
Sillitto Enterprises,  
Edinburgh, Scotland, UK  
[hsillitto@gmail.com](mailto:hsillitto@gmail.com)

James Martin  
Aerospace Corporation  
Chantilly, Virginia, USA  
[james.n.martin@aero.org](mailto:james.n.martin@aero.org)

Regina Griego  
Albuquerque, NM, USA  
[drgriego@comcast.net](mailto:drgriego@comcast.net)

Dorothy McKinney  
ConsideredThoughtfully.com  
[dorothy.mckinney@icloud.com](mailto:dorothy.mckinney@icloud.com)

Eileen Arnold  
ConsideredThoughtfully.com  
[eparold5@aol.com](mailto:eparold5@aol.com)

Patrick Godfrey  
University of Bristol, UK  
[Patrick.godfrey@systems-thinking.co.uk](mailto:Patrick.godfrey@systems-thinking.co.uk)

Dov Dori  
Technion University, Haifa, IL  
MIT, Cambridge MA, USA  
[dori@technion.ac.il](mailto:dori@technion.ac.il)

Daniel Krob  
CESAMES  
Paris, France  
[daniel.krob@cesames.net](mailto:daniel.krob@cesames.net)

Scott Jackson  
Burnham Systems  
[jackson@burnhamsystems.net](mailto:jackson@burnhamsystems.net)

Copyright © 2018 by Sillitto et al. Published and used by INCOSE with permission.

**Abstract.** We envision that Systems Engineering (SE) can be transformed into a truly transdisciplinary discipline – a foundational meta-discipline that supports and enables collaboration between all the disciplines that should be involved in conceiving, building, using and evolving a system so that it will continue to be successful and fit for purpose as time passes. SE can be applied in different ways depending on the situation and how well current SE process patterns are matched to the problem in hand. We identify four elements of this new transdisciplinary framework: SE Tenets; SE Approach; SE Process; and SE Toolbox. We suggest that the use of SE then needs to be considered in three domains: problem space, solution space, and transformation space that helps us along the development-delivery-evolution trajectory. We propose twelve SE tenets and show how they should be applied in these three domains. We perceive that even though all elements of the current SE Process can be justified in terms of the twelve tenets applied to these three domains, the current commonly used, standardized SE Process is not suitable for all situations requiring an SE Approach or an application of the SE Tenets. We claim that the framework presented in this paper can act as a unifying structure that facilitates the evolution of Systems Engineering from the current focus on a “standardized” process model suited to a particular class of problem, to a more agile and capable “transdiscipline” that will provide an enabling construct for more successful collaborations that can better deal with a wider range of complicated, complex and chaotic problem situations.

## Introduction: Reframing the Nature of Systems Engineering Practice

INCOSE (2014) set out a challenging vision for how Systems Engineering needs to develop towards and beyond 2025 to help humanity address unprecedented societal and environmental challenges as well as doing a more effective job at the engineering of complex technology-based systems. According to the New England Complex Systems Institute (NECSI) website, *NECSI has demonstrated the breakdown of conventional top-down engineering and that reliable complex systems can only be made by evolutionary dynamics and distributed collaborative design*. In a companion paper (Sillitto et al, 2018b) we demonstrate that the current INCOSE definition of Systems Engineering is not compatible with the future trajectory set out by INCOSE's 2025 Vision, and indicates how the definition needs to change. This paper describes how that change entailed in the new definition might be accomplished.

**Overall aim.** Our aim is to see how SE can be transformed into a transdiscipline – a foundational meta-discipline that supports and enables collaboration between all disciplines that need to be involved in conceiving, building, using and evolving systems that will be successful and fit for purpose. SE can be applied in different ways depending on the situation and how well matched current SE process patterns are to the problem. SE is concerned with the Problem Space, the Solution Space, and the “transformation ecosystem” responsible for Development, Delivery and Evolution (DDE) of the solution system. This ecosystem contains one or more trajectories that dictate how the solution becomes realized and continues to be relevant, and how the problem itself will change as the system and its environment co-adapt to each other.

### The starting point – a system

A system does things its parts can't do on their own (Sillitto et al, 2017; Dori & Sillitto, 2017). Thus, the added value of systems engineering is managing the relationships, interdependencies and interactions between the parts and with external entities to create the desired whole system properties and capabilities (Hitchins, 2007), so that the desired outcomes can be achieved in the problem space.

*A digression on what we mean by “capability” and “concepts”. The emerging standards on Systems Architecting frame system architecture in terms of “properties and concepts”. This use of language illustrates the difference between “architecture as a practice within SE” and “SE as a whole”. “Concepts” are very fuzzy and relate to the system itself and how it is intended to interact with the world. Capability is its ability to do something in the real world, and relates to the outcome we are trying to achieve with the system, which is determined by the stakeholders’ collectively agreed success criteria. We need to keep attention focused on the success criteria for SE and its context system, which relates to providing the means for stakeholders to achieve successful outcomes. Concepts are a means to that end. The end goal of SE is not to “create a concept”, Concepts emerge as part of the architecting process, which provides a blueprint that the rest of SE uses to systems-engineer a real-world solution that provides the capabilities required by users to achieve desired outcomes.*

*A digression on what we mean by “outcome”. Note that when we use the word “outcome” in a Systems Engineering context we do not mean “output”. This is a common misunderstanding of the real intent. Even though the dictionary says that outcome and output are synonymous, the outcome we really mean is the “result” that comes about when the system is utilized to do something.*

Most systems interact with their environment by exchanging material, energy, force and information. Thus, systems change, and are changed by, their environment; a system's effectiveness, delivered value, and fitness for purpose depend on what the system actually does, and how the system actually works, in its actual real-world operating environment.

Often, a system becomes part of a bigger system when placed in its operating environment. Thus, we must analyse the current problem situation **as a system**, and understand how the proposed new or improved system will interact with and change the rest of the “problem system” to predict its effectiveness. We usually need to think of multiple layers or levels of system, with new properties and capabilities emerging at each level. We will often be well advised to apply systems engineering at multiple interacting levels in a complex systems endeavour.

## The science, challenge and practice of systems engineering

Most engineering disciplines are underpinned by one or more branches of science. The “science” needed to underpin Systems Engineering is not yet mature. Fundamentally, we can regard “systems engineering science” as the quest for knowledge of how different sorts of systems behave under a range of circumstances (Hybertson, 2009).

The intellectual and creative challenge of systems engineering is in seeing how to apply systems knowledge and principles to understanding and framing the particular problem, and to considering alternatives and selecting and synthesising an appropriate solution concept. The challenge is both intellectual and creative because the problem space is typically too complex, and the solution space is typically too large and disjoint, to allow an exhaustive search.

The **practice** of systems engineering is concerned with both a **systemic** approach to understanding the problem, devising a solution, and understanding the interdependencies in the work to develop, deliver and evolve the solution; and a **systematic** approach to establishing objectives and success criteria, analysing and documenting the solution, predicting its effectiveness, and establishing and implementing an effective and efficient process for development, delivery and subsequent evolution.

## What is different about a Transdiscipline?

An opportunity in the environment for SE is increasing interest in the concept of **trans**-disciplinarity, as opposed to **inter**-disciplinarity. From a research perspective (Rousseau, 2015) *transdisciplinarity springs from insights about patterns that emerge only by looking across disciplines, so tell us something about the fundamental nature of the world; hence it is a powerful additional problem solving technique. GSTD [General Systems TransDisciplinarity] is about systems and is general, so it applies always and everywhere.*

Translating this thinking into a Systems Engineering context, "not just having engineers involved" is the key feature of Transdisciplinarity that makes it different (and better than) multi-disciplinary and cross-disciplinary approaches, especially for complex socio-technical challenges.

**Transdisciplinarity** connotes a strategy that crosses many disciplinary boundaries to create a holistic approach. For example, in defence and aerospace systems engineering, it is common practice to involve both operational users and specialists in human factors and psychology when formulating operational concepts, user requirements, and human-machine interface designs. In general, a transdisciplinary approach enables *inputs and participation across technical and non-technical stakeholder communities* and facilitates a systemic way of addressing a challenge.

Transdisciplinarity emphasises the benefits of bringing in people who actually live and breathe the problems trying to be solved, such as ordinary people, local politicians, businesses, social groups, etc. – in other words, ensuring that the key stakeholders come “into” the SE process. But it also entails having those outsider stakeholders help shape the process to be used for problem and solution space definition. This is because people bring their worldviews with them when they start examining what they think the problem really is, which carries over when you start trying to solve the "problems" you identified. You don't just invite them to design reviews to see if they concur or not. This is an example

of how a system, in this case Systems Engineering itself, both changes and is changed by its environment.

Such a development would provide a Systems Engineering meta-framework for integrating all the different discipline activities – technical and non-technical, from within and far beyond engineering - in a systems context.

## Four Aspects of Systems Engineering

Our latest thinking on the way forward, that would remove current barriers to SE becoming a Transdiscipline, involves recognising four aspects of SE (Fig 1):

1. some very basic and widely applicable **SE tenets** (principles and beliefs);
2. a general **SE approach** to complex and complicated problems;
3. the “**SE process**”, which we see evolving from the current SE process described in the INCOSE SE Handbook into a family of SE process models, targeted towards different system types; and
4. an **SE toolbox** of techniques and methods that are widely applicable across the spectrum.

These four aspects can be thought of as nested within each other, and all of them are nested in turn within the broader domain of systems thinking.

**Systems Thinking.** There is no single agreed definition or scope and boundary of “systems thinking”. Many different, and not necessarily compatible, systems thinking methods and approaches are used in many different sectors and domains, underpinned by different and not necessarily compatible systems worldviews. Systems Engineering only inhabits part of this wider systems landscape. Authors such as Lawson (2010) and Frank and Elata (2005) discuss what systems thinking means in a Systems Engineering context. Sillitto et al (2018a) discuss the wide variety of system worldviews held even within the INCOSE community. In the work described here, we view systems thinking and the systems sciences as a rich and diverse field from which to draw inspiration and ideas.

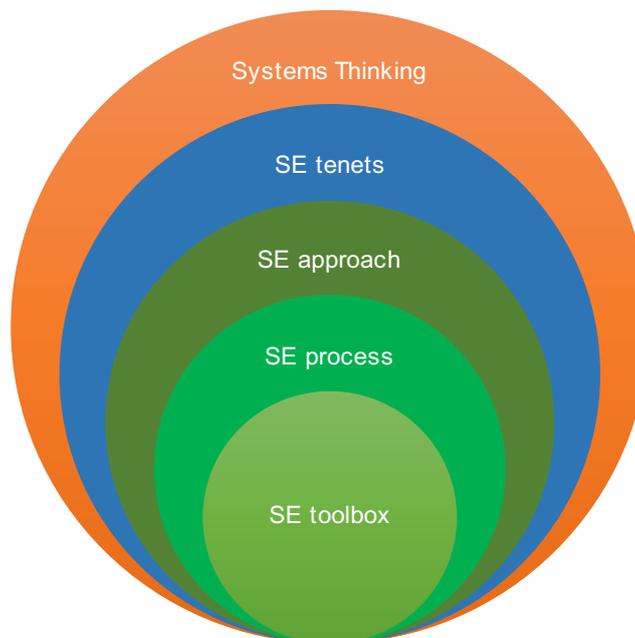


Figure 1. The four aspects of SE practice, in the wider context of Systems Thinking

**SE tenets.** Systems Engineers often talk about using a “systems approach” to work their way into a problem that seems wider or fuzzier than “normal engineering” (whatever that is). Such a “systems approach” uses selected systems principles or beliefs that are of proven value in an engineering

context and are also useful elsewhere. We will use the term “systems engineering tenets” to refer to a key set of principles and beliefs drawn from various branches of systems thinking and the systems sciences, that seem to underpin most or all of what we currently recognize as systems engineering.

**SE Approach.** The “SE Approach” would then be the deliberate use of those principles in an adaptive but systematic way to deal with the problem at hand, in a situation where an existing “SE Process” template was deemed inappropriate or inadequate. Where an existing process template fits the problem, we can (with due care) move quickly through the “SE approach” level to use the appropriate SE process template.

**SE Process.** An “SE process” is a recognized framework of activities that get used in the context of a particular SE approach suited to a certain class of problems. Currently, many systems engineers talk about “the” SE process. Kemp et al (2015) assert that the currently dominant SE process model is suitable mainly for large one-of-a-kind solutions to complicated problems, and needs to be substantially modified to deal with complex, simple and chaotic systems and situations. We agree with this analysis, and envisage that families of SE process models will be developed to cope with multiple dimensions of variation, of which we discuss three.

One dimension of variation is variety in the solution. For example, product line engineering needs a different approach from one-of-a-kind solution engineering, and socio-technical systems need a different approach from pure technology systems.

Another dimension is the nature of the problem space. Complex situations may need a “probe-sense-respond” strategy (Snowden, ....); while a chaotic situation may change quickly and often, and be devoid of repeating patterns, the only remedy in such cases being a fast and agile response.

A third dimension is the nature of the organizations assembled to develop, deliver and evolve the solution. Often such organizations are multi-national, multi-cultural and multi-disciplinary (across and beyond engineering) and the traditional SE management approach with overtones of “command and control” need to change to an explicitly collaborative or “servant leadership” model, where the emphasis is on SE supporting and enabling, rather than trying to control or manage, the other participants.

**SE Toolbox.** Many current SE tools, techniques and methods, including many of the current “support processes”, fit within the “SE Toolbox”. We don’t propose to detail the contents of the toolbox, but it would include a wide range of ways and means for analysing, understanding, explaining, measuring, structuring and formalising the problem, the solution, and the solution’s development, delivery and evolution trajectory.

## **What might be the “tenets” of systems engineering?**

A principle can be defined as a fundamental truth or proposition that serves as the foundation for a system of belief or behaviour or for a chain of reasoning (Oxford Online Dictionary). The following principles have been developed partly by considering fundamental propositions about systems and systems engineering, and partly by “reverse engineering” from accepted systems engineering good practice. It would be tempting perhaps to call them “systems principles”, but the systems field is wide and its practitioners argumentative, and it seems better to remain focused on propositions that are necessary and sufficient to justify systems engineering as it is and needs to become. The authors, after much debate and exploration of literature, identified twelve principles or tenets that seem to be core to the systems engineering approach (in the sense of fundamental propositions – the field is not yet mature enough to call them fundamental truths, tempting as this might be). These are listed below in Table 1 below, and then discussed in turn. Table 2 lists key sources for the twelve tenets and indicates how they are applied in currently published SE practice.

Table 1: The twelve proposed systems engineering tenets

<b>The Twelve Systems Engineering Tenets</b>
1) Understand what success means
2) Consider the whole problem, the whole solution and the full lifecycle
3) Understand and manage interdependencies
4) Adapt the parts to serve the purpose of the whole
5) Recognise that Systems Engineering occurs at multiple levels
6) Base decisions on evidence and reasoned judgement
7) Recognise uncertainty while managing change, risk opportunities and expectations
8) Handle structure and behaviour as two complementary aspects of any system
9) Understand and use appropriate feedback
10) Understand and manage value
11) Be both systemic and systematic
12) Respect the people

**1) Understand What Success Means.** Understand the problem and the stakeholders. Understand what success means for each stakeholder, and establish shared vision, purpose and success criteria among all stakeholders.

Establish a clear agreed objective and success criteria for the intervention system. - E.g. *“I believe that this nation should commit itself to achieving the goal, before this decade is out, of landing a man on the Moon and returning him safely to Earth”* - John F Kennedy, 25th May 1961.

Success criteria include what the system should not do. To the extent possible, unintended consequences should be anticipated and minimised. Consider alternatives, and choose the preferred concept and architecture based on how well each concept and architecture meets the success criteria.

Formally, we can say that the objective should be formulated as a name of a major function, a process that the system must deliver, the operand(s) (the objects processed by the system whose attribute values are expected to change as a result of the system's function), and the benefit created by that function.

**2) Consider the Whole Problem, the Whole Solution and the Full Lifecycle.** Systems Engineering is concerned with the whole problem and the whole solution, including how the “intervention system” will interact with its environment as part of a larger system when it is deployed, and all the enabling systems and services required to establish and maintain system effectiveness throughout its lifecycle until eventual satisfactory disposal.

We need to consider the full lifecycle of the entire solution, including all the enabling systems that go along with the system of interest.

Table 2: Sources, justification and current application of the twelve tenets

<b>The Twelve (proposed) Systems Engineering Principles or Tenets</b>	<b>Source</b>	<b>Application in current SE</b>
Understand what success means	Elliott et al (2007), principle 1: “debate, define, revise and pursue the purpose”.	Business and Mission analysis, Measures of Effectiveness, Requirements
Consider the whole problem, whole solution, and the full lifecycle	INCOSE current definition of SE. Elliott et al Principle 2, “Think Holistic”; Principle 5, “Be creative”	Lifecycle processes (ISO/IEC 15288)
Understand and manage interdependencies	Inherent in “A system does things its parts can't do on their own.”	Rechtin (1991) “the leverage in architecting is at the interfaces”
Adapt the parts to serve the purpose of the whole	Earliest identified articulation of this concept is Galen, 122-200 AD, described by Schiefsky (2007)	Hitchins (2007) see e.g. synthesis discussion p.23
Systems Engineering occurs at multiple levels	Brook et al (1998)	SEBOK – enterprise, service, SoS and Product systems engineering
Base decision on evidence and reasoned judgement	Buede (2004); article on Decision Management in <a href="http://www.sebokwiki.org">www.sebokwiki.org</a>	CMMI Decision Analysis and Resolution process
Recognise uncertainty; manage change, risks, opportunities and expectations	Blockley and Godfrey (2017), White (2006)	CMMI and ISO/IEC 15288 Risk and Opportunities Management processes
Structure and behaviour are two complementary aspects of any system	Originally Aristotle (384-322 BCE), more recently Hubke & Eder, 1988	e.g. Object process Methodology Dori (2002), Interacting Object Process Model (Blockley & Godfrey, 2017)
Understand and use appropriate Feedback	Core principle of Cybernetics, Wiener (1948)	Feedback is extensively used in system design. SE Metrics, when well used, provide feedback on process and product effectiveness.
Understand and manage Value	Oppenheim (2011): Lean SE Principle 1 – “Value” Ring (1998)	Requirements should reflect customer value.
Be both Systemic and Systematic	Combination of two principles listed by Elliott (2007) – “think holistic” and “follow a disciplined procedure”.	Current SE is argued to have become more systematic than systemic. (Hitchins 2007)
Respect the People	Elliott et al (2007): Principle 5, “Take account of the people”; Oppenheim (2011): Lean SE Principle 6, “respect for people”	Involve stakeholders in the SE process; ensure work outputs are understood by recipients; communicate effectively.

**3) Understand and manage interdependencies.** Since system level properties emerge from the interactions between the parts, and between the system and its environment, it follows that systems engineering is primarily concerned with selecting and adjusting appropriate parts for the system, and managing the interactions a) between the parts, and b) between the system and its environment, to create all and only the required system-level properties and outcomes.

In system trade-offs, it is critical to identify the key set of coupled variables that interact to constrain the solution space. For example: the classic iron triangle of cost, schedule and quality; the interaction between payload, range and speed in an aircraft; the trade-off between throughput and number of stops in a mass transit system - if you run the trains faster they can carry more passengers per hour, and the end to end train journey is faster, but to run faster they need to miss out stops, which makes some passengers' end to end journeys slower. It's important to balance the different success criteria in solution selection, rather than focus on one or a few success criteria to the exclusion of others.

Simplify and manage interfaces, both in the solution architecture and in the organisations, contracts and work breakdown structures.

**4) Adapt the parts to serve the purpose of the whole.** The aim of systems engineering is to synthesise a system that does what is needed and nothing else. To do this the parts of the system need to work harmoniously and efficiently together. To achieve harmony and efficiency we usually need to adjust both the parts and their interactions.

**5) Recognise that Systems Engineering occurs at multiple levels.** Systems Engineering may be applied recursively at multiple levels of the system. At each level we are concerned with:

- how our system contributes to the goals of the system at the level above;
- the properties our system needs to have to do that; and
- choosing and specifying the parts of our system and how they will fit together and work together to create the required system-level properties.

**6) Base decisions on evidence and reasoned judgement.** SE decisions are made based on evidence and reasoned professional judgement, not on unqualified opinion, taking into account the level of risk and uncertainty.

While fitness for purpose can only be conclusively established in operation, it is good practice to use prior knowledge, professional judgement, opinion surveys, modeling, experimentation, and incremental transition to operation to demonstrate evidence of fitness for purpose progressively from the earliest stages of the lifecycle.

In complex situations, decisions often involve making tradeoffs between the benefits to some and the losses to others. Understand which decisions need to be taken when. Take decisions at the "last responsible moment", using reasoned judgement if evidence is unavailable or inconclusive

**7) Recognise uncertainty while managing change, risks, opportunities and expectations.** We don't know everything, we can't know some things, and almost everything changes. In many situations we can't fully understand the problem until we start work - SE is a learning journey.

In a complex situation a "probe – sense – respond" strategy is required, because we can't fully understand how the new system will affect the problem situation until it is deployed and we can measure its actual effect. In chaotic situations, there is no consistent pattern, and the problem changes faster than we can write requirements to describe it.

Sometimes, the greatest benefit can be had by focusing on opportunities rather than risks.

In complex situations we need to design for adaptation and evolution through life, and use an adaptive and evolutionary approach to delivery. We also need to anticipate what people are expecting from the

effort to resolve their issues and help them adjust their expectations to be more realistic and informed by holistic understanding.

**8) Handle structure and behaviour as two complementary aspects of any system.** When analyzing a system or problem situation, it's important to understand how the structure supports and enables behavior. When designing a system, a key early step is to establish the logical architecture (describing how the system works, in terms of processes, flows and behavioral response to stimuli) and map it to the physical architecture showing the structure and interconnection of physical parts.

**9) Understand and use appropriate feedback.** Most systems have internal feedback mechanisms which maintain system stability in the face of external disruption. These feedback loops may generate unexpected and counterintuitive emergent properties, including sudden system collapse when integrity limits are exceeded. Successful system design depends on understanding the nature of feedback in the problem situation and in the system of interest, and also in the socio-political system that develops, delivers and evolves the system of interest (see below). Recognition and use of feedback is an essential skill in many applications of systems engineering.

**10) Understand and manage value.** Value is benefit at cost. If the cost incurred by creating and operating the system is higher than the benefit, the system is not viable and should not be pursued. All systems engineering effort involves cost, some of which is a "fixed overhead" for doing it at all, sometimes referred to as "the marching army cost". Because of this, a strategy of maximising value with the available resources is usually more successful in the long run than a focus on minimising cost, which frequently results in an inadequate and poorly integrated solution that is not fit for purpose and fails to meet stakeholder needs.

**11) Be both Systemic and Systematic.** The competences in SE are of two generic sorts. **Systemic** ones consider the whole problem and whole mission, and the whole system over its whole lifecycle, establishing the overall system vision to ensure **coherence**; while **systematic** ones - *done or acting according to a fixed plan or system; methodical* – underpin the systemic ones by ensuring **completeness** and **consistency**. As an example of the difference, table 3 contrasts systemic and systematic skills in two classic areas of SE, Architecture and Requirements.

Table 3: Systemic versus Systematic

Activity:	Systemic	Systematic
Architecture	Architecting a new system is a <b>systemic</b> , analytical, inventive, and creative process performed by a system architect or architecting team. It involves understanding the situation, problem and opportunities; considering a range of alternatives; choosing the best; and creating a suitable solution architecture and measures set.	Creating a model of the architecture in a particular modelling language in a particular tool is a <b>systematic</b> process. It involves making and managing a self-consistent and coherent architectural model and/or set of documents that describes the architecture as conceived by the architect(s) and shows how it captures and addresses the concerns and perspectives of all stakeholders.
Requirements	Deciding which requirements are key, establishing appropriate target ranges for their attribute values, and ensuring coherence are <b>systemic</b> activities requiring expert judgement, and a whole-system, whole-mission, whole-lifecycle perspective.	Writing a complete and consistent set of requirements that satisfies quality criteria, and tracing requirements from one level to the next, and to relevant rationale, feasibility evidence and test evidence, are <b>systematic</b> activities requiring attention to detail and a detailed understanding of the process and quality criteria.

**12) Respect the People.** In complex endeavours, success requires a variety of perspectives (discipline, cultural, gender...) and consequently a collaborative approach. In this context it is important to realise and to emphasise that SE cannot be "one approach to rule them all" but is rather a transdiscipline whose goal is to **support** the other disciplines and leverage the benefits of cultural and social diversity to ensure their outputs are effectively integrated to create a successful outcome in the form of a whole-

system solution that is fit for purpose and meets stakeholder needs in the real world. Sillitto et al (2018b) and Godfrey (2015) emphasise that this needs the systems engineering leaders and managers to adopt a collaborative leadership style to introduce the systems engineering approach as an enabling framework for successful collaboration.

### ***How do we apply the Systems Engineering Approach?***

We believe there are three domains where the SE approach should be applied:

- to understanding the problem situation;
- to the design and analysis of the solution system;
- and to the organisation and processes involved in the transformation; to be more explicit, this perspective covers conception, development, delivery and through-life evolution of the system of interest – the entire set of activities involved in transforming the problem situation using the solution system.

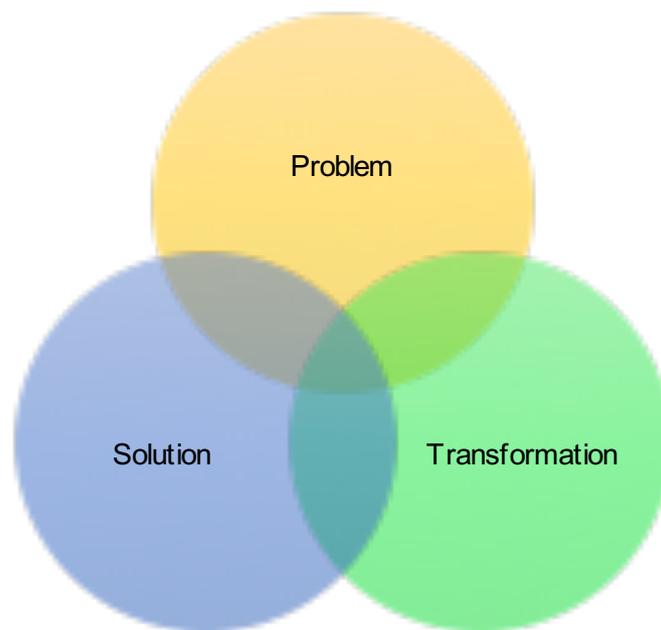


Figure 2. The SE Triad: Problem, Solution, Transformation

We illustrate this as a simple triad or trefoil diagram (Fig 2). We explicitly don't show the three as a sequential process, because in complex and chaotic situations the problem can never be “solved” by a one-shot effort. It is now the norm that the problem situation changes over time, and when a new system is deployed it further changes the problem situation, possibly generating opposition. (One situation where it is very important to realise that this may happen is when deploying Systems Engineering into an existing organisation!)

Thus we need to think about a continual [“Value Cycle” \(Ring, 1998\)](#). In the first cycle, the solution modifies the problem situation in unexpected ways, leading to unexpected effects. Subsequent “problem-solution-transformation cycles” create an evolutionary feedback loop in which the solution is adapted over time to increase and then maintain its effectiveness in the real world.

**Problem space.** In the problem space, the SE approach involves:

- viewing the problem as a system,
- understanding how the interdependencies between the elements in the problem space create the “problem symptoms”, and how the “intervention system” might alleviate the problem symptoms

- understanding stakeholder interactions and interdependencies and establishing overall agreed purpose and success criteria
- anticipating and aiming to minimise potential adverse or unintended consequences of the intervention system
- scanning for and early detection of anomalous behaviour and unintended consequences – not all can be anticipated beforehand

**Solution space.** In the solution space, the SE approach involves

- identifying potential solution approaches,
- selecting a suitable approach based on evidence and expert judgement, guided by purpose, and taking into account the levels of risk, uncertainty and change;
- defining the solution, the component parts and their properties, and the required enabling products and services to design, make, test, deploy, use, assess, support, evolve and eventually retire and dispose of the system

**Transformation space.** In development, delivery and evolution, the SE approach involves several key actions and several quite complex considerations, including the following

- situation assessment – given the nature and complexity of the problem and solution, deciding what sort of SE lifecycle approach is most appropriate and most likely to succeed;
- realising that different disciplines and organisations have different ways of working, which are efficient in isolation but may be incompatible with each other - this is an example of the systems principle that *the parts need to be adapted to serve the purpose of the whole*.
- providing leadership and management to enable the realisation and continuing success of the system;

The systems engineering process described in the INCOSE SE Handbook is a model for how the work needs to be organised and synchronised across the whole system and throughout the entire lifecycle. It aims to ensure that the different work activities are done by the right people at the right time in the right sequence, with the right information, to the right level of quality, bearing in mind the potential need for iteration and adaptation to deal with change, risks, opportunities, and expectations; and the results of the work made available and communicated to those who need it.

## **Applying the SE Tenets to each domain**

Having identified a set of twelve Systems Engineering Principles and three Domains for the Systems Engineering Approach, we can now show more precisely – in Tables 4-6 - how to apply the principles to each domain.

(We should make it clear that the current SE process model set out in the INCOSE SE Handbook does apply the twelve SE Tenets we have listed. However, it builds a complicated process framework around them, and Kemp et al (2015) argue that this complicated process framework is only well adapted to complicated problems involving complicated engineering solutions. They illustrate the need to adapt the SE process for complex systems, and highlight the potential cost and risk of failure if the chosen SE process and approach are not properly matched to the problem.)

Table 4: Applying SE Tenets to the Problem Domain

Tenet	Problem Domain application
Understand what success means	Establish a clear definition of purpose and success criteria. Understand what success means for each stakeholder, and establish shared vision, purpose and success criteria among all stakeholders.
Whole problem whole system whole lifecycle	View the problem as a system. Recognise that introducing a new system that solves or suppresses the existing problem may create new ones. Situation may include technical, societal and naturally occurring systems.
Understand and manage interdependencies	Understand how the elements of the problem interact, and how the new system will interact with and change the environment. Understand the relationships and interdependencies between stakeholders, and how that affects their views of purpose and success criteria. Understand how different success criteria interact to constrain or expand the solution space.
Adapt the parts to serve the purpose of the whole	It may be possible to solve the problem by changing the existing operational practices rather than adding a new system.
Systems Engineering at multiple levels	Each layer of the solution provides the problem for the layers below.
Base decisions on evidence and reasoned judgement	Maintain clear line of sight from all decisions to purpose and success criteria. Characterise the problem – e.g. is it simple, chaotic, complicated or complex? – and use an appropriate approach (e.g. for complex – “probe sense respond”). Where “evidence” is not available or inconclusive, use reasoned judgement to make the best decisions possible at the time.
Uncertainty, change, risks, opportunities and expectations	Understand dynamics of change in problem space. Accept we don’t know everything, we can’t know some things, and almost everything changes. Realise that sometimes we can’t fully understand the problem until we start work. Manage stakeholder expectations to be realistic..
Structure and behaviour	Look for structures in the problem space that cause the problematic behaviours. Look for leverage points that will allow an intervention system to alleviate the problem while minimizing adverse consequences.
Feedback	Look for the feedback mechanisms that keep the problem going in spite of attempts to solve it. Understand how the feedback mechanisms in the problem space might be exploited to amplify the effects of a successful intervention.
Value	Understand stakeholders’ perceptions of value. Seek opportunities to manage and maximize value.
Systemic and Systematic	Use appropriate shared model-building methods and tools to enable the transdisciplinary group to frame the problem, for example Soft Systems Methodology (SSM), Interpretive Structural Modelling (ISM), Causal Loop and System Dynamic Modelling (CLM/SDM).
Respect the people	Ensure you understand who is affected by the problem and possibly who is benefiting from it. Bring those people together and facilitate in such a way that they all have equal opportunity to express those opinions, and that those opinions are valued and considered.

Table 5: Applying SE Tenets to the Solution Domain

Tenet	Solution Domain Application
Understand what success means	Establish a clear agreed objective and success criteria for the intervention system. Understand how every part of the solution contributes to fitness for purpose.
Whole problem whole system whole lifecycle	The solution may be a product, product line, service or enterprise. Consider all aspects of the solution including enabling systems and services required for a successful system that is fit for purpose.
Understand and manage interdependencies	Systems Engineering focuses, <b>not</b> on the design of the parts, but on the required properties of the parts and on how they need to interact to create the required properties, capabilities, behaviours and outcomes of the whole solution. Understand how key attributes of the solution interact to shape the “fitness landscape” of potential solutions. Simplify and manage interfaces.
Adapt the parts to serve the purpose of the whole	Parts may need to be sub-optimised, and off-the-shelf parts reconfigured, so that they work harmoniously as part of the system.
Systems Engineering at multiple levels	Several layers of the solution may have parts that are worth treating as systems in their own right.
Base decisions on evidence and reasoned judgement	Select a suitable approach based on evidence, guided by purpose, and taking into account the levels of risk, uncertainty and change. Take decisions at the “last responsible moment”, using reasoned judgement if evidence is not available or inconclusive.
Uncertainty, change, risks, opportunities and expectations	Keep options open until at least one is definitely suitable. Architect to allow opportunities to be exploited, and to minimise cost and disruption of likely late changes. Design for adaptability and evolutionary learning. Manage participants’ expectations to be realistic.
Structure and behaviour	Apply equal attention to behavioural/logical view of the system as to the physical and structural view (develop both logical and physical architectures).
Feedback	Understand and carefully manage the feedback loops in the solution and in the interaction between the solution and its environment so that the system will behave in an intuitively acceptable way both in envisaged and (to the greatest extent possible) in unexpected situations.
Value	Focus on maximising value. When working to meet a cost target, make sure the value proposition of the solution remains compelling.
Systemic and Systematic	Keep attending both to the big picture (“are we building the right system, will it solve the problem?”) and the detail (“will it work as expected in all relevant situations?”)
Respect the people	Consider the implications of different cultures involved in using and supporting the system. Realise that the technical part of the system is usually there to help the people using it, not the other way round. Make sure that the technical and process solutions fit the mental models and reasonable expectations of the intended users, operators and other stakeholders. Make sure that unintended negative consequences for all stakeholders are avoided or mitigated to the greatest extent possible.

Table 6: Applying SE Tenets to the Transformation Domain

<b>Tenet</b>	<b>Transformation domain application</b>
Understand what success means	Recognise the tension between fulfilling contract and solving problem. Choose the preferred concept and architecture based on how well each concept and architecture meets the success criteria. Provide management and leadership to enable the realisation and continuing success of the system
Whole problem whole system whole lifecycle	Consider all aspects of delivery processes through the whole lifecycle required to transform an idea of a solution into a successful system that is fit for purpose. Recognise that in some situations you can't establish all requirements until the system is operational. Choose an appropriate architecture and lifecycle model to fit problem and solution.
Understand and manage interdependencies	Offer the systems engineering process as a model for how the work needs to be organised and synchronised across the whole system and throughout the entire lifecycle. The goal is that the different work activities are done by the right people at the right time in the right sequence, with the right information, to the right level of quality, bearing in mind the potential need for iteration to deal with risk, uncertainty and change, and the results of the work made available and communicated to those who need it. In complex endeavours, success requires a diversity of perspectives (discipline, cultural, gender...) and consequently a collaborative approach. Simplify and manage interfaces between organisations and work packages.
Adapt the parts to serve the purpose of the whole	Different disciplines and organisations have different ways of working, which are efficient in isolation but may be incompatible with each other. How to get them working harmoniously in the right sequence to achieve the purpose of the whole?
SE at multiple levels	SE is applied at "whole solution" level, and also in selected lower level systems, subsystems and services.
Base decisions on evidence and reasoned judgement	While fitness for purpose can only be conclusively established in operation, SE aims to build evidence of fitness for purpose progressively from the earliest stages of the lifecycle, using prior knowledge, professional judgement, opinion surveys, modeling, experimentation, and incremental transition to operations.
Uncertainty, change, risks, opportunities and expectations	Measure and report on uncertainty (lack of knowledge) and change dynamics. Use appropriate incremental commitment approaches to make progress while learning about the problem, containing risk, exploiting opportunities, and keeping options open. SE is a learning journey and may require an adaptive and evolutionary approach to delivery.
Structure and behaviour	Ensure there is a clear understanding of who does what, why, where, when and in what sequence, and how.
Feedback	Use appropriate elements of the Systems Engineering Toolbox (modeling, metrics, experimentation) to provide early feedback about the fitness for purpose of the solution, and effectiveness of the delivery work. Avoid triggering the "organizational immune response" when introducing SE.
Value	Ensure that work done adds value.
Systemic and Systematic	Lifecycles may be transient. It is possible that a particular development stage or element requires an agile approach as requirements are not fully understood, whilst another development stage or element is well understood and is more plan driven. Multiple developments may also need to interact. This must be understood and agreed by all parties, and revisited at appropriate points.
Respect the people	Understand the information flow. Who needs what information, in what format, by when and what for to ensure that no team is left frustrated. Ensure the development is appropriately synchronised so one team moving forward does not force a decision on another. Minimise the development 'stovepipes' and maximise collaborative working.

## **Is the current SE Process always fit for purpose?**

The current SE process model set out in the INCOSE SE Handbook applies the twelve SE Tenets we have listed. However, it builds a complicated process framework around them, and Kemp et al (2015) argue that this complicated process framework is only well adapted to complicated problems involving complicated engineering solutions. They illustrate the need to adapt the SE process for complex systems, and highlight the potential cost and risk of failure if the chosen SE process and approach are not properly matched to the problem.

The new challenge facing SE is spelled out by INCOSE (2014), and can be summarised as a transformation of the traditional SE process towards a situation where we can develop systems in which we cannot control all aspects of behavior. This includes socio-technical systems, complex systems and some Systems of Systems. Various INCOSE Working groups, including Systems Science, Natural Systems, SoS, and the Complex Systems Working Groups, are grappling with these issues, and this is the motivation for the effort to redefine SE (Sillitto et al, 2018b). INCOSE may need to recognize different kinds of SE Processes for different kinds of systems, which might need separate handbooks. These types include

- a) technical systems we design to be controlled,
- b) systems that we can influence but not control (e.g socio-technical, natural, complex),
- c) systems where we can only attempt to understand.

Each of these by their nature demand different approaches in both SE analysis and SE management, but share common systems and SE principles – a way of thinking and approaching systems problems.

## **Summary and conclusions - What Next?**

We envision SE becoming a TRANSDISCIPLINE – a foundational meta-discipline that supports and enables collaboration between all disciplines that need to be involved in conceiving, building, using and evolving systems that will be successful and fit for purpose. SE can be applied in different ways depending on the situation and how well matched current SE process patterns are to the problem. SE is concerned with the Problem Space, the Solution Space, and the Transformation ecosystem responsible for Development, Delivery and Evolution (DDE) of the solution system.

We outlined twelve SE Tenets, and showed how they apply to the three domains of problem, solution, and transformation.

We believe that these Twelve Tenets, mapped to the three domains, can be used as the basis for a generic SE Approach that can be applied to any problem situation where engineered systems are part of the problem, or part of the potential solution. The appropriate approach can be used as the basis for a formalized process whose nature depends on the levels of risk, uncertainty, opportunity and change present in the situation. We think that the current “standard” SE process will evolve into a family of SE processes adapted to different classes of problem, noting that these classes of problem will be characterized not by the application domain (defence, aerospace, automotive etc) but by the nature of the “system problem” (levels of risk, uncertainty, opportunity and change). Such Systems Engineering processes would be truly transdisciplinary and trans-domain, would enable collaboration between all who need to be involved in creating successful complex systems, and would allow the SE community make a major and effective contribution to addressing the challenges set out in the INCOSE 2025 Vision.

## References

- Blockley and Godfrey (2017) *Doing it Differently – systems for rethinking infrastructure*, Thomas Telford press, 2017
- Brook, Arnold, Stevens and Jackson; *Systems Engineering – coping with complexity* – Prentice Hall, 1998
- Buede, D.M. (2004). "[On Trade Studies](#)." Proceedings of the 14th Annual International Council on Systems Engineering International Symposium, 20-24 June, 2004, Toulouse, France.
- Dori and Sillitto (2017) *Defining System – an ontological approach*, Systems Engineering Journal, May 2017
- Elliott et al, 2007 – *Creating Systems that Work – principles of engineering systems for the 21<sup>st</sup> Century*, Royal Academy of Engineering, London, June 2007
- Frank M & Elata D (2005) Developing the Capacity for Engineering Systems Thinking (CEST) of Freshman Engineering Students, Systems Engineering Journal, VI 8 No 2, April 2005
- Godfrey P, *Building a Technical Leadership Model*, INCOSE International Symposium, Edinburgh, July 2016.
- Hitchins D (2007) *Systems Engineering – a 21<sup>st</sup> Century Systems Methodology*, Wiley, 2007
- Kemp D, Beasley R and Williams S (2015) “Suits you sir! -- choosing the right style of SE before tailoring to fit”, INCOSE International Symposium, 2015,
- Hybertson (2009) *Model oriented Systems Engineering Science – a unifying framework for traditional and complex systems* CRC Press 2009
- INCOSE (2014) *A World in Motion: Systems Engineering Vision – 2025*, INCOSE 2014
- Lawson H (2010) *A journey through the systems landscape*, College Publications, London, 2010
- Oppenheim (2011), *Lean for Systems Engineering – with lean enablers for systems engineering*, Wiley, 2011
- Ring J 1998 - *A value-seeking approach to the engineering of systems*, IEEE International Conference on Systems, Man, and Cybernetics, 1998.
- Schiefsky 2007 - [Galen's Teleology and Functional Explanation](#). Oxford Studies in Ancient Philosophy. 2007;33 :369-400.
- Sillitto H, Arnold E, Dori D, Godfrey P, Griego R, Jackson S, Krob D, Mckinney D, Martin J, (2018a) *What do we mean by “System”?* – *System beliefs and worldviews in the INCOSE community* – INCOSE International Symposium, Washington, July 2018
- Sillitto H, Arnold E, Dori D, Godfrey P, Griego R, Jackson S, Krob D, Mckinney D, Martin J, (2018b) *Redefining Systems Engineering for the 21<sup>st</sup> Century*, INCOSE International Symposium, Washington, July 2018
- Sillitto H, Arnold E, Dori D, Godfrey P, Griego R, Jackson S, Krob D, Mckinney D, Martin J, (2017), *Defining System – a comprehensive approach*, INCOSE International Symposium, Adelaide, July 2017
- White, B. E. (2006), 5.1.1 Enterprise Opportunity and Risk. INCOSE International Symposium, 2006
- CMMI: Capability Maturity Model – Integrated, SEI
- ISO/IEC/IEEE. 2015. *Systems and Software Engineering -- System Life Cycle Processes*. Geneva, Switzerland: International Organisation for Standardisation / International Electrotechnical Commissions / Institute of Electrical and Electronics Engineers. ISO/IEC/IEEE 15288:2015.
- SEBOK – Systems Engineering Body of Knowledge – [www.sebokwiki.org](http://www.sebokwiki.org)

## Biography

Hillary Sillitto held senior systems engineering and architecting positions in Thales and the UK MoD, and is now pursuing a variety of academic and professional interests. His book “Architecting systems – concepts, principles and practice” was published in October 2014. He contributed to the INCOSE SE Handbook and the SEBOK. He is an INCOSE Fellow, and Omega Alpha Association member.

James Martin is an enterprise architect and systems engineer affiliated with The Aerospace Corporation. He was a key author for the SE Body of Knowledge (SEBOK), led the working group developing ANSI/EIA 632, and the INCOSE Standards Technical Committee. He previously worked for Raytheon Systems Company and AT&T Bell Labs. His book, *Systems Engineering Guidebook*, was published by CRC Press. Dr. Martin is an INCOSE Fellow and received the Founders Award.

Dorothy McKinney is an INCOSE Fellow, serving on the INCOSE Definition Team. She has over 45 years of aerospace and research experience, working over 34 years at Lockheed Martin and heritage companies. She served as an adjunct professor at San Jose and Portland State Universities. She now heads ConsideredThoughtfully Inc., providing online mentoring for professionals.

Professor Dov Dori heads the Enterprise System Modeling Laboratory at the Faculty of Industrial Engineering and Management, Technion, Israel. He is Fellow of IEEE, INCOSE, and IAPR and a Visiting Professor at MIT. His research interests include MBSE and conceptual modeling of complex systems, and systems biology. He invented and developed Object-Process Methodology (OPM), adopted as ISO 19450, authored over 300 publications and mentored over 50 graduate students.

Regina M. Griego, Ph.D. is an independent Systems Consultant and retired as a Distinguished R&D Systems Engineer at Sandia National Laboratories after 20 years of service. She has over 30 years of experience leading multi-agency and multidisciplinary teams in various domains to deliver systems and develop organizational capability. She is a teacher, mentor, and coach and recognized for her research and ability to elicit a common conceptual basis for realizing solutions. Regina is an INCOSE Fellow, past Technical Director, and the Enchantment Chapter Founding President.

Eileen P. Arnold, Fellow of INCOSE, ESEP, and PMI PMP worked 35+ years as a system engineer in weapons systems and commercial and Business/Regional aircraft. She currently works as the co-founder of a Start Up company, ConsideredThoughtfully, Inc., developing virtual professional mentoring applications. She received the MFESTS Charles W. Britzius Distinguished Engineer Award. She has been an active INCOSE and IEEE volunteer and author since 1996.

Patrick Godfrey has been civil engineering practitioner for 50 years. He was a Director of Halcrow for 30 years, Professor of Engineering Systems at the University of Bristol for 10, and is now an Emeritus Professor. He believes systems thinking and learning skills are two sides of the same coin required to deliver Vision 2025. He is a Fellow of: the Royal Academy of Engineers, INCOSE, the Institution of Civil Engineers, the Energy Institute and Honorary Fellow of the Institute of Actuaries.

Daniel Krob is currently Institute Professor at Ecole Polytechnique and chairman of CESAMES, the French leading company in systems architecting and engineering. He authored more than 100 papers, 6 books and 3 patents in areas including combinatorics, computer science, communications and systems modeling. He managed many training, consulting and transformation missions in systems engineering within major industries among aerospace, automotive, bank, energy and high tech.

Scott Jackson is an independent consultant and researcher. His book *Systems Engineering for Commercial Aircraft* (2<sup>nd</sup> Edition) was published by Ashgate Ltd in the UK in 2015 (available in both English and Chinese). He also wrote *Architecting Resilient Systems: Accident Avoidance, Survival, and Recovery from Disruptions*, (Wiley 2010). He taught graduate level courses in systems engineering at USC, and now assists aircraft companies around the world in systems engineering.