SAND2021-14055C

Collaborative Systems Thinking Culture:

A Path to Success for Complex Projects

Author1
Aff. Addr. Place
Aff. Addr. Place2
Phone
email

Author2
Aff. Addr. Place
Aff. Addr. Place2
Phone
email

Author3
Aff. Addr. Place
Aff. Addr. Place2
Phone
email

Author4
Aff. Addr. Place
Aff. Addr. Place2
Phone
email

Author5
Aff. Addr. Place
Aff. Addr. Place2
Phone
email

Commented [AF1]: We shall think on a good title… Topics: SEE17 Technical Leadership and SEE06 Systems Thinking. More? Please, check the Excel file uploaded by Mickael

Commented [RW2]: Tried a different title.

Commented [LJ3]: Avoid acronym CST - this is commonly thought of as Critical System Thinking.

Commented [LJ4]: Two potential options - Systems Thinking Culture or Collaborative Systems Thinking Culture.

Commented [AF5]: We will include as many authors as needed

Contents

Abstract

The world is filled with hard, complex problems seeking solutions. To make these often daunting problems more manageable to solve,both a mindset shift, and key candidate methodologies centered around a collaborative systems thinking culture are proposed. The idea is to introduce not just some collaborative practices, or systems-thinking approaches. Rather, the proposal for solving the tough problems – complex problems that basic approaches do not seem to solve – is to move an organization beyond basic techniques into a culture that has as its core a collaborative and systems-thinking theme. This paper will present a an introduction to what a collaborative systems thinking culture (CSTC) is and looks like. The paper starts with exploring the state of the practice, presents the mindset change involved with systems thinking, propose that a collaborative approach is a part of this shift, and then conclude with the 7 phases that the reader can introduce into their organization to realize some of the benefits.

The fundamental premise is: amongst a team, group or an organization, everyone brings their own perspective. A collaborative environment seeks out these perspectives (as appropriate) - and forms a way to get key perspectives out into the common "pool of knowledge" [VitalSmarts reference, Crucial Conversations]. The systems-thinking enters when there is outreach to more and more different (yet related) perspectives. The goal is to get a nearly-complete set of related perspectives, timed appropriately based on the project phase. Getting these out leads to smoother programs and better products. In pursuing this goal, we consider competencies, future steps in improvement, and new approaches to solve systems-level problems. What this paper proposes, is that the most effective way to develop this habit of appropriate and properly-timed inclusivity, is to "bake" this mindset into the culture of how engineering work is performed in an organization: to grow a Collaborative, Systems-Thinking Culture (CSTC). This approach will prepare the technical teams for more complex problems, with increasingly numerous and complex interfaces – without relying on vast corporate engineering procedures to drive collaborative behavior by fiat or inspire a bigger-picture systems approach to design and development through written guidance or lessons-learned descriptions in a database. Technical (and non-technical) teams can solve problems, finish projects on time and develop good systems engineering habits organically without being forced (at least not at first!). While changing culture in an organization is hard – and the CSTC approach will deny both the team and customers some "early progress" in the form of a slower-appearing project ramp-up, the benefits will be worth the wait. The benefits of working on the culture, and fostering a more organic growth and change, are that the changes to how work is performed, tend to stick. This saves time and money by avoiding rework, and also prepares teams for even more complex problems downstream.

Commented [RW6]: There is a lot of first person, which I know is controversial in technical papers. Need to revisit case and person.

Commented [LJ7]: In general, it is best practice to not use 1st person narration in a formal paper. I have attempted to revise to remove "we"

Commented [RW8R8]: If this were a thought-piece, would we be OK? Let's revisit. For brevity I did leave in a lot of first person for now to get the thoghts out.

Introduction and Background

Authors: @Raymond Wolfgang @Anabel Fraga @Emma-Rose Tildesley

Reviewers: Jay Patel, @Jason Baker

We live and work in a complex, interdependent world – and will continue to do so!

The world is getting more complex. More humans are connected to more devices which are connected to more networks; these interactions are being captured (and sold) as data is stored, analyzed, and presented in multiple forms of media. Such data can be (and is) archived and processed (Artificial Intelligence) by data houses, or shared with millions of people around the world in an instant. This hyper-connectivity is in part driving the complexity of the problems we as systems engineers are asked to solve. How did this happen? When exploring the question of "Why do things become more

complex?" W. Brian Arthur posed that "complexity tends to increase as function, and modifications are added to a system to break through limitations, handle exceptional circumstances or adapt to a world itself [which is] more complex". If we apply this thinking to the modern world, we can observe new function and modifications added to existing systems every day, cars are on their way to being driverless, computers are on their way to interacting directly with our neural pathways, people are interacting virtually, products are being made out of recycled and waste materials, computers are getting smaller and more powerful. All this leads to far more complex – and interesting – environments in which engineers try to add value and perform their work.

At the same time, organizations have been evolving and changing. In some organizations, traditional hierarchical structure is becoming redundant, and being replaced with a fully-matrixed, project-based informal authority network. Some organizations have even removed "middle management" (although in some instances, they are quickly reinstalled as guidance and supervision of the work is still needed). What is certainly true is that more and more sophisticated capabilities are being outsourced across the globe, where some roles no longer exist whilst other new roles have arisen; again, while not the subject of this paper, AI plays a role in this. On top of all this, customers' needs and values have transformed, workers interact remotely virtually from all over the globe, digital upskilling is constant, and artificial intelligence has not just shaped the labor market – but also customer's expectations of products and of interactivity. How can a modern organization continue to operate effectively in this complex modern world? Especially since budgets and timelines have not expanded to accommodate for a thorough and measured transformation initiative within the company to support all these new aspects of making and delivering a technical product or service. Interesting times indeed! Fortunately, there are ways engineering organizations can adapt to both survive, and to thrive.

Here is where adopting a collaborative culture, with an emphasis on systems thinking along with open communication, can help an organization navigate a landscape that (even before Covid) was becoming decentralized, digital-heavy and drifting further from traditional organizational models. How does a company do this? According to Conway, the first attempts would be based on a company's internal structure (which in part can drive or cement its internal culture). Conway's law suggests that "organizations which design systems... are constrained to produce designs which are copies of the communication structures of those organizations" (Conway, 1968). This is not a bad prognosis, nor is it abnormal considering the nature of human organizations. In fact, this is a great place to start – the organizations structure and communications culture – for any change initiative. Adopting a more systems-thinking culture in the organization, that takes root organically, qualifies as a "change initiative". The answer, the paper proposes, is not simply to work the traditional model – assigning tasks to individual teams, allocating design responsibility do specialized departments, then integrating at the very end – more aggressively. Giving everyone overtime, may yield some short-term results, but if the old paradigms are worked harder the project may simply reach integration problems more quickly. As Demming is alleged to have said, "It is not enough to do your best – first you must know what to do, then do your best." He is also quoted as "It is not necessary to change; survival is not mandatory." The cultural shift this paper proposes is not necessarily easy, but can boost an organization's success in tacking modern problems.

The thesis of this paper is to adapt to the changing environment mentioned above an organization must make a shift to a collaborative-based systems-mindset culture to adapt fully to not only the current business and engineering environment but the one 10 years in the future. While neither collaborative cultures nor systems-thinking are new, the idea of introducing both into the communications structure and business processes of an organization has not been as explored. The proposal is that by introducing both from a perspective of organizational culture will result in a less constrained, more innovative and ultimately successful organization as measured by cost, scope, schedule, quality and customer satisfaction performance. This blend is called a "Collaborative

Commented [ET9]: http://www2.csudh.edu/gsterling/complex.pdf

Commented [LJ10R10]: Be sure to include in the references section at the end of the paper.

Commented [RW11]: May need to explain this.

Commented [RW12]: PMP reference? This is from the PMBOK, of a few years ago.

Systems-Thinking Culture (CSTC). Definitions of this CSTC will be presented and a methodology on how to achieve this will be proposed.

This paper uses four pillars, throughout the different sections, to motivate adoption of a CSTC to solve an organizations toughest technical problems:

1. Identifying the problem: desire to solve more complex problems to cost/schedule
2. Systems-Thinking as a prerequesite
3. Collaboration with healthy team dynamics as key – the CSTC
4. Complexity model to classify problems to determine strategy and tactics (Cynefin)

The four are progressive, and the paper approaches them in sequence – if we aggressively collaborate before adopting a systems-thinking mindset our projects will drift. Introducing a classification model before establishing the CSTC becomes an academic exercise. After this foundation is created, we present the process the reader can use to move to a CSTC and – with the complexity model to help classify our projects – can start efficiently tackling far more sophisticated development efforts and analysis work.

First, what problem is CSTC trying to solve? The following is a list of problems, anectotally, that the community may be familiar with:

- Components of your system or product were built to the correct print, but the interface had changed at the next level of assembly - forcing rework or possibly a redesign
- Cost estimates for verification and validation activities were known to be much higher than projected – but this information had not reached the systems integration team until a major design review in front of the customer, causing an awkward and credibility-reducing request for more time and money.
- Designers, equipped with a decent "first draft" of the requirements, promptly worked on design prototypes to show the team, management, and the customer progress was being made. Future requirements updates were ignored, resulting in a stale prototype once the metal was cut.
- The project management (master schedule) effort and engineering work drifted apart midway through the project. While engineering work was being performed, and PM metrics collected, they were disjointed. Soon, engineers were working "off the cuff", focusing on an initial design, producing a prototype or performing testing according to their judgement and experience. Too much time was spent on design before checking back with schedule or requirements – causing a time-consuming effort to rebaseline.
- Scope was added by the design engineers, because the extra features were "low-hanging fruit"; the result was that other tasks were not completed or key testing activites were not started on time.

If any of this looks or sounds familiar, and has caused cost overruns, schedule slips or simply led to wasted time, effort and money – then a CSTC may help mititage or solve these problems, and improve overall project performance. While any new system takes work and organizational commitment to set up, and a formal organizational-change process (outside the scope of this paper) is recommended [reference] for this or any change initiative - some of the benefits from the process the reader can look forward to are:

- Interfaces are worked early, and taken seriously.
- Less cost overall (from reduced rework) – with more accurate cost/schedule estimates along the way for both management and the customer.
- Fewer technical and programmatic (cost/schedule) surprises downstream, late in the project. This also reduces rework.
- Customer thinking included in the project flow, in a formal way
    - Customer environment, and expectations, baked into the systems engineering
    - The product is much more likely to feel more organic to the customer, by using this approach. Much reduced chance of "this is the wrong rock" syndrome, where the product or system meets the letter of the Statement of Work or other requirements document but does not satisfy (let alone delight) the customer. The original need or pain-point that drove the work in the first place, is only partially fixed.
- Designers still focus on their task, but they get the right input earlier on and at appropriate intervals.
    - One part of the CSTC is creating a robust and well-staffed systems integration team. Once stood up, this group provides both technical input to the designers, and regular contact with the customer to facilitate prompt and fresh information flow. Note that here, and throughout this document, Agile processes may still be used; the type of interation with the customer on an agile project would be in addition to the "systems – level" interation described here.
    - If designers are too collaborative, they may take longer to produce the design work since they would be attending additional meetings. A collaborative systems engineering culture and approach, does not mean designers regularly meet with the customer – unless it is part of an agile system, or otherwise orchestrated so that the design team continues to receive pertinent and fresh information without being overwhelmed with interaction.
    - This will lead to an overall design benefit – prototype cycles will be more closely aligned to original needs, therefore the number of design cycles could reduce. Here is where model based tools, including Model-Based Systems Engineering tools, can come into play. Design cycles produce a more accurate 'first cut'.

Many of these ideas are not new – and it will benefit to first describe the state of the art and practice of collaborative-focused, systems engineering practice and culture, before moving onto the proposed process and approach. The paper then moves to the tailor-able process to introduce this type of mindset and culture into an organization to begin to solve some of the problems listed above. The goal is to move into enjoying some of the benefits not just to cost and schedule, but also to increased quality and customer satisfaction. As with any process, the reader is encouraged to tailor it to their needs, and to the change-ability of the organization. Too much change too quickly – especially cultural change – can wind up killing the initiative.

State of the Art: Collaborative Systems Engineering in Organizations

Authors: @Mickaël BOUYAUD @Erika Palmer @Jean DUPREZ

Reviewers: @Joseph Gaskell @Alex Deng @Raymond Wolfgang

A Good Head Start

Many organizations already utilize systems-thinking, and have parts of their culture very aligned with a collaborative environment. They have structured technical teams into cross-discipline groups, "Integrated Product Teams" which have representatives from (as an example) specialty engineering (electrical, mechanical), project management, quality, safety, security, risk – along with the team of design and systems engineers. Here the intent is to smooth out the communication between areas of expertise and different parts of the organization by including representatives in team meetings and project flow. This can include physically co-locating members of the team in the same office, to improve synergy; co-location is not mandatory, and sometimes not possible as the last 18 months have shown. Even if this is not the team model, with enough experience in the organization team leaders and members will have worked with others in different groups over their career to build a network of personal acquaintances and friendships. "I know Jane, over in corporate risk – we should tell her about this design change, especially since it might affect our testing schedule" might be heard in the hallways of an organization, where the informal communication flow is healthy and robust. This is where an organization with high turnover can struggle with being efficient. A culture that supports this type of cross-organization informal communication, without fear of "trespassing" or butting into another's "swimlane" will support this type interaction. This informal approach can answer questions very quickly.

Definitions and Context

Systems thinking has arisen not only as an approach to embrace complexity, but is considered a critical tool for organizations in adapting to rising complexity. Donella Meadows (2008) in her book "Thinking in Systems: A Primer" explains that systems thinking can transcend disciplines and culture with the potential to cross-connect historical boundaries and specialties. What this means is that organizations are interconnected with and cut across social and sociotechnical systems (i.e., disciplines and cultures) and can evolve and reshape system structures over time.

Systems thinking is "utilizing modal elements to consider the componential, relational, contextual, and dynamic elements of the system of interest" (Davidz 2006). This holds true for products and systems – as well as the organizations developing and building them. Systems Thinking can help us to extend our range of observation, to deeper and better analyze how we perceive the system, to questions what we have achieved, verifying consistency of outcomes (going back to previous steps, if needed), and thus to give us the opportunity to better understand and master the systems, while mitigating risks of negative bias. It is similar to "big-picture" thinking, but also includes a grasp of how the different elements of this big picture interact, in both normal and out-of-normal conditions.

Systems-thinking is very related to systems-science. Systems science is a scientific field that is rooted in the theoretical study of systems, i.e. systems theory. It is an interdisciplinary field, in that it seeks to understand complex systems and how they connect in the physical, natural, social and psychological domains. Fundamental systems science concepts that are applied in systems thinking practice include for example:

Commented [LJ13]:

Commented [LJ14]:

Commented [RW15]: US Dept. of Defense reference?

Commented [RW16]: This may need some clarification.

Commented [RW17]: Add reference, to book in office by CRC press.

Commented [RW18]: Moved this up, from the end of the section.

- The interconnectedness of the social, environmental, economic and cultural complexity of systems
- The drivers and root causes of system behavior
- Transitions from linear production to circular production systems
- Resilience to system shocks
- Emergent system behavior
- Synthesized stakeholder knowledge to build and strengthen sustainable system relationships
- System wholeness

Systems thinking is the practical application of systems science concepts. In the System Thinking disciplines, Systems Engineering is classified into Hard System Thinking. Cf. M. C. Jackson. Some claim that Hard System Thinking is worthful for technical systems but reaches the limits of the methodology once it shall address complex, social-technical systems which still are difficult to represent with mathematical models. This limits its application fields of studied systems and the coverage of the System Engineering for being a full transversal discipline. This argument has been seen as valid and some works have been successfully conducted into INCOSE and its chapters that demonstrate progress on dealing with human and social-technical systems. The reader is referred to active INCOSE working groups that deal with these subjects for more interaction.

The other half of the CSTC is collaboration. We are not alone, and certainly few companies allow individuals, no matter how gifted, to work alone consistently. We are people and work with people. We are a team and part of a team. Can you imagine, if a company set up a sophisticated Model Based Systems Engineering (MBSE) environment, fully embracing digital twin concepts in design and production – but only one top engineer, plus the administrator, had access? The designs created by the "lone wolf" - were then rolled out to the world like Moses and the 10 commandments. That is asking for disaster, or barring that, at least the rest of the team quicly reverting to more traditional, pencil/paper/MS Office methods to perform systems engineering. This is why the collaboration piece is so relevant.

When combined with systems-thinking, CSTC helps us extend our previous ideas of team behavior, the interrelationships, and dynamics of the group. This improves teamwork and provides space for each member of the team to improve their own performance and the improvement of team performance. The better prepared each of the members is to adopt this mindset, the more effectively the team interacts, the team members will understand *how* the team is operating in real-time, and by extension the more effectively the team wll be understood by the other parts of the organization. Ihn most organizations this means the team will more quickly move to accomplish its goal. In most cases, this means the team accomplishes the scoped technical task, with a wise use of resources (including labor time), on time and on schedule. As individuals, our capabilities are limited - even for the very senior engineers; no one can keep all the details of today's complex projects "in their head". To go beyond our limits, in a way that drives to a realistic budget and schedule, we need to collaborate with others. This is why the collaboration piece cannot be split from the systems-thinking part.

Aspects to Consider, before blending CSTC into an Organizational Culture

A definition for collaborative systems thinking has been derived and proposed as "an emergent behavior of teams resulting from the interactions of the team members and utilizing a variety of thinking styles, design processes, tools, and communication media to consider the system, its components, interrelationships, context and dynamics toward executing systems design" by C.T. Lamb and D. H. Rhodes. It requires an understanding of systems thinking and organizational culture.

Commented [AD19]: We need to differentiate the difference between systems science and systems thinking, and the evolution of systems thinking?

Commented [LJ20R20]: This is still missing - add this analysis to conclude the section?

Commented [MB21]: Is there a systems science as such? Are you refering to ISSS works? Then we need to take care from the editor point of view if what we are saying is aligned. For example, Cynefin methodology is not an application of system science.

Commented [LJ22R22]: Agree - need to drop a reference that is listed at the end of the paper.

Commented [RN23]: Unless we mean something specific by "system engineering" as distinct from "systems engineering," we should use the INCOSE standard term "systems engineering." Similar recommendation for "systems thinking" which is the standard used in SEBoK.

Commented [MB24R24]: indeed. Corrected

Commented [RW25]: Add reference – Lenconi, 5 dysfuntions of a team?

Commented [AF26]: Emma, it is related to the "why" collaborative...

Commented [ND27R27]: I added a definition of Collaborative System Thinking reference 2. I also added the reference to the files folder

Commented [RW28]: This section seems redundant with the previous paragraphs. Can we look to reduce here? There are some good, unique ideas here still.

Commented [LJ29]: Missing a citation.

Commented [ND30R30]: reference 2? Lamb, C.T and Rhodes, D.H., "System Thinking as an Emergent Team Property: On going Research into Enablers and Barriers to Team-Level System Thinking", IEEE International Systems Conference, IEEE, New York, 2008.

Organizations are social systems, and their systems of symbolic meaning inherent in organizations can be described as cultural capital (a type of cultural system, a subgroup of social systems). Collaborative systems thinking is the emergent behavior of organizational systems and their cultural capital. If we as an organization (or leader in such) want to lead our group to a collaborative, systems-mindset – we certainly need to be aware and work with the organizational culture *as it currently is*, not necessarily as we want it to be.

What is culture?

There are countless definitions of culture, though when applied in a systems context, it is helpful to think of culture as "cultural capital". Cultural capital (Bourdieu, 1986) is a symbolic form of capital produced by labor that is not economic. In an organization, for example, people "produce" capital as they work to establish routines and structures that hold symbolic value. Someone who is seen as a good employee would be one who works within the boundaries of structures and routines of what is symbolically considered good within the organization. People who hold a lot of cultural capital within an organization are those that people seek out as mentors and are often found (though not necessarily) in leadership positions, formal or informal. Cultural capital with other forms of symbolic capital such as social capital – where one is placed in the network of organizational status. This can be a formal position, such as a leadership position/title or an informal position (the "go-to guy/gal").

The culture of an organization represents the "way of life" that distinguishes the organization from others. It involves how the organization sees and describes itself and its mission – both tightly tied to organizational identity. In some cases, the organization has a defined *mission* and/or *vision* that it uses to define itself and portray both internally and externally its unique character and its goals. However, this tells only part of the story. An organization's mission and vision largely define what it strives to be, reflecting its perceived self-image. However, the real underlying culture of the organization is largely embedded within the history of its organizational structure and in the norms of behavior that have been established over the course of many years.

Motivation to address culture as the starting point

An organization's body of policy and procedure —the standards for process execution and assessment—are perhaps the most visible of these, being captured explicitly in forms that are formally promulgated as requirements or guidance. However, much of the organization's inherent culture is undocumented and is maintained and perpetuated by habit and custom. Organizational and individual behavior are largely guided by incentives, both explicit and implicit, and these incentives can be enablers or impediments to realizing organizational agility. An established organization has at least modestly produced results, or else it would not have lasted long. The ways in which results have been produced and incentivized form the culture – and since past success is the best protector of the status quo, these ways need to be examined by Systems Engineering and System Thinking approaches before any attempt to change them starts.

The Link to Systems Engineering

INCOSE promotes a System Engineering framework containing tools to develop, maintain, manage elements of knowledge, a culture, of the system into the producing organization. This knowledge can serve other engineering disciplines for taking decisions, and designing products. Methodology is derived from hard sciences and mathematical models ensure the scientific proof of the results. As a non-exhaustive list, it encompasses modeling of system concepts, functions and activities during all operational phases, semantic federation with ontologies and knowledge management process (cf. Rousseau), use of languages (UML, SYSML, …), even how to liaise with stakeholders of the organization (business, projects, management, …). It aims at developing a global comprehension of

Commented [AF31]: Erika could help here!!!

Commented [EP32R32]: On it :-)

Commented [EP33R32]: I could go on, but let's see how the section takes shape from other collaborators first

Commented [LJ34R32]: @Mickaël to lead organizing of this content for this section to close it out.

Commented [EP35]: Bourdieu, P. (1986). "The Forms of Capital. Readings in Economic Sociology" J. Richardson (Ed.), "Handbook of Theory and Research for the Sociology of Education, vols. 241–258", Greenwood Press, New York.

Commented [LJ36R36]: @Leema to add to the references.

Commented [AD37]: need to elaborate more to include Systems Thinking Culture?

Commented [LJ38R38]: Agree - missing a definition of "Systems Thinking Culture" at the end of this section before talking about the proposed process. Perhaps add this definition to conclude the section?

Commented [LJ39R38]: @Mickaël to follow up with section authors.

Commented [MB40R38]: @Alex and editors, section is about collaborative system thinking, not culture. But I agree it is a good idee to introduce the definition. Any advice?

Commented [AD41R38]: @Mickaël got your point, is there anything from SEBOK we can used?

Commented [AD42R38]:

the system inside its environment. This can be the foundation for the development of a culture inside an organization. Knowledge of this culture, from at least a point of situational awareness, is crucial to linking thes concept together: a collaborative approach, systems-thinking, and the systems engineering needed to execute a program and have a product at the end of it.

Integrating the knowledge of each of the disciplines involved in the elaboration or evolution of a system using the system engineering methodology by itself can become huge and unwieldy. To close this gap, Systems Engineering requires to integrates itself with mor cultural domains. This is where the tieback to systems-thinking comes in.

In the System Thinking disciplines, Systems Engineering is classified into Hard System Thinking. Cf. M. C. Jackson. Some claim that Hard System Thinking is worthful for technical systems but reaches the limits of the methodology once it addresses complex, social-technical systems which still are difficult to represent with mathematical models. This limits its application fields of studied systems and the coverage of the System Engineering for being a full transversal discipline. This argument has been seen as valid and some works have been successfully conducted into INCOSE and its chapters that demonstrate progress on dealing with human and social-technical systems. Again, the reader is referred to the collection of INCOSE working groups, symposia [IS 2021] as well as INSIGHT magazine [July 2021] for engagement and information on systems thinking as applied to social systems; otherwise social systems per se are ouside the scope of this paper – except to note that a development organization, is its own social system – and sits in the middle of yet another, larger social system.

Commented [RN43]: Unless we mean something specific by "system engineering" as distinct from "systems engineering," we should use the INCOSE standard term "systems engineering." Similar recommendation for "systems thinking" which is the standard used in SEBoK.

Commented [MB44R44]: indeed. Corrected

The action step at this point, is to blend the idea of a CSTC, with the systems engineering that needs to be performed to product a product or system in a “real world” organization. Again this is not a new idea – as other works developed in other System Thinking forums already promote the importance of the System Engineer to collaborate with other disciplines. Collaboration in particular with social systems, ones that are themselves System Thinking cultures although they may not call themselves that name, is cited as particularly important in building a shared culture. The work in [3] relates to a project elaborated with a system thinking approach which permitted the sustainable purification of a polluted river. Author calls for System Engineering to enhance and to collaborate with other disciplines for managing complex projects. It comes with the building of a shared comprehension and culture from different domains incl. social, political and economic. This is sometimes called Soft System Methodology, or Critical System Thinking. Frameworks like Cynefin give us a model to name and classify problems that have this social system aspect, or to address the layers of complexity with social-technical systems. If we are attempting to blend CSTC into our own organizational culture, and we want those changes to stick and become organic to the working culture, that by definition is a social-system endeavor.

Commented [RW45]:

This paper has discussed systems-thinking, collaboration, and systems engineering as it relates to what work needs to be performed in the first place. The last aspect to address before introducing a process, is what does this paper mean, by “complex process” or “a complex system”? Here is where the Cynefin model adds value. The Cynefin model allows the reader to classify different types of complexity, to separate complexity from merely “complicated”, and to help realize when our problem may be large, but fundamentally “simple”. This saves us time by helping us identify the strategy we need to solve our problem (i.e. the customer’s problem). If we find we do have a genuinely ‘complex’ problem, we can then start with the amount of CSTC that is needed to address the situation. Going through the Cynefin model also forces us to think about the problem up front – and when in the “complex” space, further motivates the CSTC approach. Using a CSTC, takes work – and is not something we would necessarily use for every project, although it’s helpful to have such a culture in general. No matter what answer we get from Cynefin – we will have an answer on how to approach

the solution since CSTC has already been introduced; Cynefin will help with the tailoring. A description and discussion on Cynefin follows.

Cynefin framework and Collaborative System ThinkingThis section introduces the fourth pillar – the complexity model. This model now becomes most useful after we have established 1) the problem, 2) systems-thinking as a concept, and 3) collaboration, to form the groundwork for the CSTC. A model to describe complexity will benefit us at this point, in part to motivate the need for a collaborative approach and mindset to solve the problems that more and more are far more complex than traditional industrial-model based approaches to engineering can solve.

The Cynefin Framework

The Cynefin framework ([6] and [7]) has been created in 1999 by Dave Snowden, to help leaders to identify how they perceive situations and to analyzing behavior and make decisions. It is a conceptual "sense-making" framework that has been used in many sectors to help giving awareness of what is really complex and what is not, and to respond accordingly. In this case, "respond accordingly" involves creating the systems-engineering based strategy and tactics to achive our goal: develop a product, solve a problem, or perform customer-requested analysis. One goal is not to waste energy in overthinking the routine – making things overly-complex, while avoiding trying to make the complex fit into standard solutions (as tempting as this may be), resulting in unexpected negative effects. Some projects cannot be completed with high quality, using systems engineering methods that worked in the late 1990's for instance. The basic Cynefin model is in Figure 1.

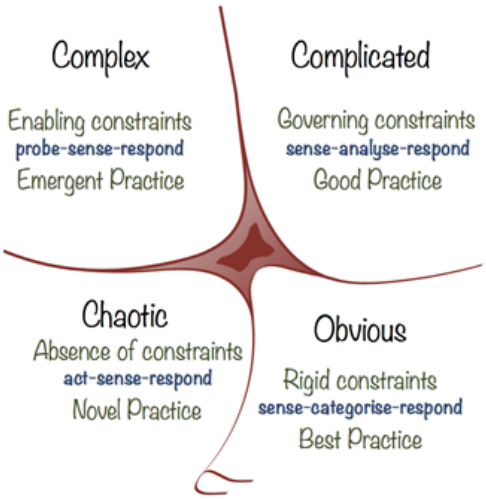

Figure 1. Cynefin Framework

The Cynefin framework characterizes the systems complexity looking at causes & effects of different parts of the system, as they are known (not all may be). It identifies 5 domains:

The domain in the center is called 1) disorder domain, where we don't know which of the other domains apply. It is not uncommon to start here. The 2) Obvious domain is associated with "ordered" systems, when cause and effect are known, predictable, or can be discovered. This "clear" domain (formerly called "simple" and "obvious") represents the "known knowns". This means that there are rules in place, the situation is stable, and the relationship between cause and effect is clear. Once facts

Commented [RW46]: Consider reducing the Cynefin description. We just need the basics.

Commented [LJ47R47]: @Raymond please work with @Jean to get this to where it needs to be.

Commented [AD48R47]: Agree, need to articulate the 'systems thinking' behind Cynefin

Commented [RW49R47]: 9/24: action is to identify if this part is in the right place. Maybe move earlier? Perhaps to motivate what we mean by 'Complex…"

Commented [RW50R47]: Added text, to explain why this was after the CSTC work.

Commented [AD51]: What is Goal I

Commented [LJ52R52]: @Alex discuss with @Jean for clarification.

Commented [AD53R52]: Closed, as it has been updated

Commented [AD54R52]: Closed, as it has been updated

Commented [AD55R52]:

established ("sense") and the situation identified (“categorize”), a decision-making solution (or process, to get the solution) can be derived from rules or best practices (“respond”). Some very large projects, may still be in this realm.

The 3) complicated domain consists of the "known unknowns". The discovery of relationships between cause and effect requires analysis or expertise. In the complicated domain, there are a range of right answers. The goal is to identify good operating practices from the assessment and analysis of facts. Although this is not as straightforward as being in the Obvious domain, the team may still be able to apply already known and reusable solutions. “Complicated” (per Cynefin) problesm can still benefit from application of clear and repeatable processes, relying on well establish workflows, rules and constraints to follow. The results may not be as predictiable, but “best practices” and good program hygiene still apply. The Obvious and Complicated domains are sometimes called the Ordered domains.

The 2 domains on the left are "unordered": cause and effect can be deduced only with hindsight or not at all. In the complex domain, there are known-unknowns but we are less certain of what we may be missing. There is still enough structure to do both a ‘probe-and-response’ analysis, and to apply the general practices applied to Complicated problems to the problem space. In the final domain, the 5) chaotic domain, cause and effect are unclear and unstable. Events in this domain are "too confusing to wait for a knowledge-based response".0 The short-term goal is to act to get stability and reach at least the complex domain, where the problem becomes more tractable. The chaotic domain represents the "unknown unknowns": cause and effect can only be deduced in retrospect, and there are no right answers. The best we can do is probe, with experiments that may or may not pass, and respond to the results; identifying emerging patterns which can both help prevent future crises and discern new opportunities. With enough probe-and-response, and with some luck, the problem can move into being merely “complex”. Knowing *which* regieme our project starts out, will help us pick the most useful short-term strategy to get started. Applying probe-and-response to Obvious problems is a waste of effort. Applying the standard, well-tested development procdures to Chaotic or even Complex problems will be an exercise in frustration – or at best, mildly informative. More information on what consitutues obvious, complicated, complex, chaotic or unknown can be found in the references cited above or online.

Commented [RW56]: Did this come from Snowden, or the Cynefin model? Quote?

Commented [RW57]: Add ref? To youtube videos of Snowden?

Clockwise Drift

Commented [AD58]: Suggest to have one paragraph for each domain, and the name of domain is not consistent

Commented [RW59R59]: Cleaned this up, worked to eliminate duplicate terms.

As knowledge increases, there is a "clockwise drift" starting from the lower left, from chaotic through complex up over to (merely) complicated and (ideally) back to simple; providing an efficient application of well establish processes and workflows, allowing reusing and sharing solutions and best practices. Depending on the situation, it may be desireable to move a problem that starts out as chaotic through this clockwise motion. Moving clockwise involves gaining knowledge of the problem space, identifying what is known vs. unknown, and searching for and finding candidate solutions or techniques that can be useful. Such knowledge increase can be done by analysis of course – but capturing key aspects of stakeholders’ concerns may also help move from chaos to merely complexity. What is the customer’s pain point? This can narrow down the problem and invite creative – and perhaps already established – solutions. Once impactful causes and effects identified, they can be explored, capturing associated dependencies. Especially when starting with chaotic problems, a useful early goal can be to build a model of the system (or mental representation) allowing to make emerging complicated representations from a pool of uncertainty. In addition, if clear rules & principles can be captured based on what is known – this forms a starting point. Soon, structured and formalized observed dependencies, can inspire and point to cause and effect relationships; expanding

this knowledge and understanding, may lead to predicting in a sufficiently simple way what would happen if we apply our known development and problem-solving practices, allowing to support clear decision making and to identify solution techniques. Soon (and with some luck) we may move a once-chaotic problem into the obvious realm where the freshly-developed solution approaches may be repeated.

To do so, such "clockwise drift" relies on our ability to embrace all impactful aspects of the system, while reducing the inherent system complexity to what is relevant to a solution (or final product or system). It leads to consider assumptions that we need to be clearly aware of, including interface assumptions. In collecing this information, and these assumptions – and to clarify assumptions into known practice – a systems-thinking mindset, combined with a collaborative development environment and culture (yes, this should sound familiar) makes this process efficient. For this reason, we introduce the CSTC before the model – since a CSTC will help us drive this clockwise drift.

Contiuous Flow Back and Forth

This 'clockwise drift' is not in one direction, as situations can easily move clockwise to more complication or complexity. When scope changes, or stakeholder needs or concerns change, our previous assumptions and models may no longer be relevant or sufficient. This would be "counter-clockwise movement", and add to the risk, for unpredictable results and potentially to chaotic situations. A project should expect both the "clockwise drift" and the identification of "counter-clockwise movements"; what can help both identify backwards (counter-clockwise) drift and prevent it, these three points are suggested:

1. Entertain hypotheses about assumptions throughout project development
2. Keep a learning culture in the team and organization. Encourage curiosity
3. When new ideas do come up, especially a concern about counter-clockwise movement – validate these ideas, and take into account all associated concerns around a maximum of impacted disciplines.

As highlighted by [8], this means to look for the involvement of relevant group(s) of people and to provide efficient collaboration frameworks, with diversity and an "clean slate" mentality as key tenets. It implies we need on continuously engage other parties, bringing as much as possible different perspectives and viewpoints, ensuring we are taking a holistic view and avoiding risk of siloing or groupthink. While anything can be done to excess, what this means practically is that we do not just interact with a host of SMEs at the beginning of the project, get their advice, then not contact them again for 3 years until we are stumped during testing with no obvious path to fix our almost-fully built system.

Commented [RW60]: ?? Spirit of an open mind?

The Cynefin framework is so necessary because there are both complicated and complex problems to be solved in our engineering and leadership work – and to solve these problems, with or without the Cynefin model, taking a collaborative approach is a must – as is changing our engineering mindset to adopt a systems mindset. Namely, to naturally consider all the possible inputs to our system, consider a wide range of perspectives during development and examine a wide range of potential impacts and influences. With this CTSC groundwork and basis – operating around the Cynefin model becomes not a dreaded chore but a natural flow: a flow to realize our product, create our system or solve the problem assigned.

Role of Formal Modelling

Formal modelling, including the use of both "mindmap" tools (miro) and model-based systems engineering (MBSE) tools can do two key tasks as we work complex projects:

1. Help us move clockwise though the Cynefin model – and help slow down (if not prevent) counterclockwise drift by capturing knowledge about a system or problem space in one place – and in such a way that it is easily editable and able to be updated.
2. Increase our knowledge of a problem space or candidate design (with interfaces) – in such a way newcomers to a project can quickly absorb the collective knowledge (again, in one location).

Much of this relies on observation (of the real system or problem) and on (mental) modeling of cause-and-effect relationships. This can get our concerns down on paper or in the model. This in turn facilitates the ability to address different perspectives, either increase our knowledge (domain(s) of expertise), skills, etc..) or invite in those with different expert-level expertise to see our progress and mind-map. Either way, more minds are approaching the same problem, which – with dialog and a culture of collaboration – can facilitate the work of solving the problem at hand quickly, and in a way that remains aware of the interfaces. How models accelerate this process – is that is presents an SME a unified definition of the problem or proposed solution; all SMEs brought in, see the same visual and get the same description. Another advantage of formal models is that interfaces may be added and detailed as they are discovered. Models have their own pitfalls, and the reader is encouraged to validate their models as possible; however, often a model must only be as accurate as needed to make the next few program decisions or perform elimination of candidate archtiectures that clearly will not work. Models can then be refined, as project resources permit, later in the flow when different decisions are needed.

Commented [AD61]: Need to tweak the language, for example , no "we". And do we have reference for this section?

The goal remains to structure either the project, organization or both to allow an efficient collaborative system thinking environment. We propose that this will drive better systems thinking emerging from the interactions – good, bad or confrontational – of the individuals involved. This will lead to not only better products but also the ability to adapt to disruptive events (again, Covid as an example). The latter part of this paper focuses on how to promote and organize such interaction (and confrontation) in a way that is useful, but also safe enough for employees to avoid a terminally polite environment absent of conflict, but also of bold progress, needed accountability or in the worst-case, an absence of working products delivered on time and to budget. The next section outlines such a process, to bring an organization into this new space or to refine on the progress already made in moving to this collaborative and systems-focused "big picture" environment – the CSTC. Once the CSTC is in place, the other steps (Cynefin, project work) can proceed efficiently.

Final Thoughts before Implemeting a CSTC-based Change Initiative.

Systems Thinking can help us to extend our range of observation, to deeper and better analyze how we perceive the system, to questions what we have achieved, verifying consistency of outcomes (going back to previous steps, if needed), and thus to give us the opportunity to better understand and master the systems, while mitigating risks of negative bias. This is a tall order even for the most experienced engineer or leader; as individual, our capabilities are limited. To go beyond our limits, we need to collaborate with others; this of course drives the 'collaborative' in collaborative systems thinking! Combining the two, and driving this into the organizational culture, gives birth to the CSTC – the enabler to the tactical work of solving very complex problems.

With an aggressive move to online platforms such as Zoom during the Covid period, this cultural aspect becomes even more critical, since we must now do our jobs without the normal person-to-person interaction. It is hard enough to change culture without everyone on Zoom; it is much harder still, once most of us are physically remote. Change is still needed, as the problems, and the opportunities for innovation an creativity – are increasing regardless. INCOSE Past President Garry Roedler provides a quote attributed to Jack Welch, former CEO of General Electric. "If the rate of change on the outside exceeds the rate of change on the inside, the end is near." This is very true for industry working in the competitive marketplace. It is also true for government engineering organizations – and even INCOSE, as the former President Roedler cited the quote to support to the idea that INCOSE's "rate of change must increase to match the rate of change in industry and the rapidly evolving technology universe." Despite the trials a change initiative will bring, the choice to proceed has to be and stay, 'Yes'. A few points to keep in mind, for this journey:

Change Initiative Tips

- Whenever adopting a new process, organizational change theory applies. These adaptations are not necessarily expensive – but may require some time (socialization) to start the shift in culture and approach.

- Once some grass-roots momentum achieved, the effort needs executive leadership buy-in. It is recommended to have the new process – at least the new openness of communication between departments – be management-driven; this may require a top-down approach to drive infrastructure changes including the adoption and IT-support for new tools. It is likely a hybrid approach will be most effective: bottom-up to build grass roots, then leading to top-down, to finish enrolling the rest of the organization.

- Acknowledge and commit to the financial and business investment – employee training, purchase of and training for new tools, and perhaps obtaining schedule relief with current customers to allow staff to work on and develop the new process. An initiative rolled out, without giving staff or supervisor-level management time to implement and learn, is destined for failure. If the investment will pay off, and an upfront investment in tools and people should be expected and budgeted for in the corporate planning process. The hope is that such efforts already have corporate support as part of a corporate-level continuous improvement culture. At the least, management should be incentivized for creative solutions to complex problems, instead of judged purely on short-term operational performance.

- There should be (if not, we suggest starting) a strategic thinking group at an organization larger than a few employees, but not so large that internal conflict slows it down. This group would ideally be identifying the complex, future problems and opportunities the corporation may or will face. Such a group should focus on more than just where to spend Research & Development money – and would be a great place to start initiatives to move to a CST culture organization-wide. They may even have already started similar efforts, but call it something different. No matter what its name, creating an organization, small as it may be, where such an initiatve can be birthed and driven in its early stages, can take the load off of executives for this early staff-level leadership. The initiative would have a 'home'.

An Identity Shift

If the desire is to implement a CSTC in a group, from a small team to a large corporate division, the remaining question becomes – how do we, as systems engineers and leaders, move individual

Commented [RW62]: Can we cite this quote? Or, perhaps remove quotes and paraphrase.

Commented [MB63R63]: no reco. I let the decision to editors.

Commented [RW64]: Do we want to include this, re: INCOSE? Or keep it general, that organizations need to stay nimble.

Commented [LJ65R65]: Review with larger team. Author is open to group consensus.

Commented [RW66]: Moved this here, from a few pages back. Seemed like a good ending note before the process.

interaction, legitimate ambition, and current healthy confrontation, to further and deeper collaboration and systems-thinking on a larger scale? How would we even establish a timeline for such a shift – assuming that not all individuals will "see the CSTC light" at the same time, or ever? When do we "carry the wounded but shoot the stragglers?" The answer is to help individuals change their identity – how they see themselves and the stories they tell – to align with this new CSTC. As the reader continues, they are encouraged to place this identity question at the forefront, both for the reader and their group. For some, the identity as "worker-engineer" may need to shift. A management shift could be from responsibility to maintain funding for all their people, to a mnore organic, abundance-based, "common good" focus – and having the larger success be the metric by which they judge themselves. If we want to lead our organizations in evolving to new and more effective ways of doing business, we too will need to evolve into new ways of thinking and acting, of moving through the world. If not, we may find ourselves perfectly trained for yesterday's way of doing business - iow left behind. The rest of this paper takes into account the mindset shift, and proposes a way to make such a change in our organization, and even ourselves. Only now that we have laid the foundation – on the CSTC, the complexity models we will need, and the concepts of mindset and identity, can we now move to our proposed process.

Cultivating a collaborative system thinking culture: A proposed process for organizations

Authors: @Natalie Davila-Rendon @Maria Romero @Jean DUPREZ @Joseph Gaskell @Erika Palmer (??)

Reviewers: @Ryan Noguchi @Emma-Rose Tildesley

This section proposes a method and process for cultivating a collaborative systems thinking culture, and centers around four concepts:

1. Training and Execution
2. The Team
3. Enablers and Barriers
4. Phases to Implementation

A Healty Head Start: SE Training and Execution

Proper Training is a Must

Part of the solution is to teach systems thinking in the corporate or organizational engineering curriculum – and better yet, to encourage students to take critical or systemic thinking classes while at college or university. Also recommended, formally or informally, are courses in non-engineering fields – politics, policy – to obtain the valuable thinking in these "non-engineering" areas. A liberal-arts approach to engineering education is a valid thought stream. An awareness, and acknowledgement of how widely different fields can interrelate, or at least have something to offer, becomes more valuable in systems thinking as one career progresses. There is a tradeoff between getting the specialized engineering knowledge, which is important for any engineering career - and being broad enough to appreciate not just collaboration but also to facilitate a systems-thinking approach and orientation.

Commented [RW67]: Heard this from Ralph Samspon, a frequent INCOSE contributor and speaker (Siemens). Not sure if he got it from somewhere.

Commented [AF68]: Jean, you included this ideas, interesting!!! I can collaborate but please lead us on this part. Is it ok for you to extend it? It must be extended on a chapter, only introduced here… right?

Commented [LJ69R69]: The concepts listed are good - could we potentially illustrate an example from one of them? That could make the paper stronger.

Commented [RW70]: Need to work on the transition, from state of art/practice to this section.

Commented [MB71R71]: see above

Commented [AF72]: Companies, organizations or corporation. We shall use the same during the paper...

Commented [LJ73]: @Natalie Davila-Rendon

Commented [LJ74]: @Maria Romero

Commented [LJ75]: @Jean DUPREZ

Commented [LJ76]: @Joseph Gaskell

Commented [LJ77]: @Erika Palmer

Commented [LJ78]: @Ryan Noguchi

Commented [LJ79]: @Emma-Rose Tildesley

Commented [RW80]: Some discussion, of including David Long's IS 21 talk, on adding several of his "V"s and "T"s as they apply – such as vulnerability, and trust needed to become deeply collaborative.

The Basics of Systems Engineering must be in place.

Knowledge of course is not enough to run a robust and successful technical organization. Organizations still need to practice rigorous configuration management, records management and change control, along with traditional systems engineering activities such as needs elicitation, requirements management and verification and validation (V&V). Strong systems engineering, combined with traditional project management including tracking risk, provides a powerful head start in driving to a collaborative systems mindset baked into the culture. For this paper and the proposed process, it is assumed that these "basics" are in place for both systems engineering and project management.

This Includes Interfaces

These "basics" include interface management. Recent (and not so recent) engineering catastrophe examples were found to have interface development as a flaw – from the units conversion error for an early US Mars Rover (REF), to the struggle pilots had with the Boeing 737 Max Maneuvering Characteristics Augmentation System (MCAS) unit [IEEE ref?]. Thorough interface management and engineering – which systems engineering prescribes – may have caught some of these errors. Here advanced modeling tools - including Model Based Systems Engineering (MBSE) tools – can help in performing these and other technical aspects of systems engineering. If interface management is weak, and systems rigor practiced only intermittently, the ideas in this paper will help but it is proposed that the project has larger issues than the lack of a CSTC.

Commented [RW81]: References needed

Commented [ND82R82]: the reference is in the files and added to the Reference section. Correct bibliography method need to be reviewed.

The Team

A collaborative system thinking culture, as it shows up in team dynamics, is defined in reference [2] as "an emergent behavior of teams resulting from the interactions of team members and utilizing a variety of thinking styles, design processes, tools, and communication media to consider the system its components, interrelationships, context and dynamics towards executing system design".

Commented [ND83]: I found this definition on a reference that could be done earlier than this section.

Successful collaborative system thinking culture teams per reference [1] have in common a team structure, experience and a team culture that binds them or that is common to all the members. This consists of three facets:

- Team structure - provides the hierarchy of knowledge, communication and leadership necessary to shape the team. Included are the Subject Matter Experts (SME)s, functional individuals with understanding of the technical and social aspects of the projects and design/system developers.
- Experience -provides the team with individuals that have worked on similar projects and can transfer that knowledge to other team members.
- Team culture, or "rules of the road" – this would provide a (hopefully) safe environment in which team members feel comfortable to bring their ideas and be heard.

Experience as stated per reference [1] is an important contributor to collaborative systems thinking while team diversity is perceived as an enabler of creativity. Therefore, measures of team diversity and experience can be of great relevance to collaborative systems thinking. With measures for team diversity and experience will enable the definition of a set of team metrics measuring both team diversity and past experiences. There is certainly overlap between the two – different team members have different experiences over their careers, which leads to diverse opions. Since each members experience set is unique, hence diverse, the two ideas cannot be truly separated. A list of considerations for both diversity and experience is below.

Commented [RN84]: It isn't clear to me that these should really be distinct. Diversity enables multiple perspectives to be brought to bear. One of the many dimensions of diversity that are relevant in most contexts is experience. A diversity of experiences enables lessons learned and best practices from the past to be exploited. Someone with 20 years in a field doesn't have more *experience* than someone with 10 years; rather, they have more *experiences* that can be tapped and learned from. These *experiences* are what matter, as they are unique. Experience is not a commodity.

Team Diversity and Experience

- Degree Concentration and Discipline are indicators of the type and variety of specialized knowledge on a team.
- Job role (sometimes reflected in title or level of responsibility) is an indication of the types of functional roles represented on a team.
- Social Styles or Personality information (Myers-Briggs, DiSC) can give an indication of team heterogeneity from a personality standpoint.
- Team Roles, and whether basic roles are filled, show how well balanced a team is from a functional/execution perspective.
- Team Tenure: The relative length of time team members have been together is an indicator of team maturity and how long teams have had to form transactive memory.
- Individual Systems Thinking Capability: If any team members have formal systems engineering degrees, or formal systems-thinking training or problem solving training (TRIZ, etc.).
- Level of Education- The number of advanced degrees on a team can be indicative of the depth of knowledge represented.
- Corporate and Industry Tenure: The number of years spent in the industry and with the current company are proxies for depth of experience and familiarity with corporate procedures.
- Experience with Past Similar Programs: The number of past similar programs worked is a direct indicator of the breadth of experience represented on the team. This can also determine where on the Cynefin chart the project starts out at, and how quickly the clockwise motion can occur.
- General "seasoning" of the staff. In many organizations, the majority of engineers are middle-aged or older. This represents a substantial amount of collective expertise and experience, although can be an indicator of change-resistance. While knowledge-preservation per se is outside the scope of this paper – the fact that many of the experienced staff can leave for retirement in 10-20 years, is part of the overall organization landscape which should be considered in any organization-wide change initiative.

Commented [ND85]: why just the aerospace industry? can this be more general?

CSTC cultures, like systems thinking at the individual level, deal with managing complexity, understanding interactions and interdependencies, and handling cross-disciplinary, or multi-disciplinary knowledge—the traits of systems thinking in engineers. For an ideal CSTC these traits are not necessarily handled by one individual, but emerge through the interactions of the team, or a team of teams. Group interactions are influenced by an organization's culture, team norms, a team's physical environment, and established engineering practices and processes. Teams operate within a culture.

Commented [MB86]: CST is already known for "Critical System Thinking".

One organizational note: As experienced engineers retire, the industry will lose a disproportionate number of its systems thinkers, and lose some diversity since seasoned engineers bring unique experiences and insight to the table. This demographic situation, combined with the lack of formal systems-thinking in much of the engineering curriculum, forms a need to accelerate the development of systems thinkers and find new ways to leverage the benefits of systems thinking. It is expected that insight into team dynamics in the context of a CSTC along with standard HR and staffing attention will help to alleviate this problem.

Enablers and Barriers to a CSTC

This paper strives to lay out a thorough foundation and motivation for a CSTC. The last part of this (extensive) background is to share some enables and barriers to implementing a CSTC in an attempt to equip the reader with as much collected insight as possible for success. Most of these are part of communication and mindset, the third concept weaved through the document.

The right mindset, mentioned earlier, along with the ability and willingness to communicate thoroughly and effectively both individually and as a team, form the third concept mentioned at the start of this section. Also, it is proposed that the process alone – without the background, motivation, or context – is not enough to inspire either the reader, or their colleagues, to undergo the hard work of change.

Enabler: Effective communication is a necessary condition for CSTC

Communication among engineers is not limited to the written and spoken word. Part of good communication in a design team is the use of sketches, drawings, mathematical equations and models. While the use of computer modeling tools may be called out in standard processes, informal sketching is very important for the creative process during early design and to help team members share ideas with one another – especially in "blue sky" or brainstorming sessions. Even late in the program, during root-cause analyses, many types of communication methods are used. While improving communication is beyond the scope of this paper, as a skill this dhould be in every systems engineer's toolbox.

Commented [ND87]: should we call this brainstorming

Enabler: Ability to engage in divergent and convergent thinking

Engineers excel at convergent thinking—beginning with a problem and finding a solution. Divergent thinking begins with a requirement, or need, and asks questions to explore the design space and to generate a large number of design possibilities. The challenge is in fostering open and critical discussion of design alternatives during the divergent phase without premature convergent thinking. Both skills are important.

Enabler: A Product/Results Focus

A clear direction and commitment to the end product is seen as an enabler for better communication, willingness to cross discipline boundaries, and ability to make tradeoffs that benefit the end system, rather than a single component or function. Nothing drives invention and creativity like a looming deadline!

Enabler: role of "Requirements Owner"

The reader is referred to the INCOSE Systems Engineering Handbook for a good description of team roles on an engineering project, but one additional role that is releveant to CSTC is "requirement owner". This person – usually but not always someone with an engineering background – is responsible for the definition, maturation, changes to, implementation (design), test and validation of this requirement over the product lifecycle. The nature of this role, demands the engineer cross-cut with the project management team, testing, quality, and the other specialties. This is another way that a collaborative systems-thinking culture can be orchestrated into the company culture simply by how the engineering work is assigned, organized and tracked. Certainly having requriements owners is not mandatory for a CSTC, but by the role's nature, it can help.

Enabler: Team internal and external situational awareness

CST is about identifying and leveraging interactions, interfaces and cross disciplinary knowledge. Team awareness is an individual trait that indicates awareness of what others on the team are working on and also what others on the team know. This knowledge, when universally held, enables team members to preemptively share information with those who need it and better coordinate efforts toward improving a system design. High-level knowledge of what other teams are responsible for and can do is also helpful – and can facilitate finding the right SME on short notice.

Barrier: The 'hero' culture, or other unhelpful incentive structures

While the role of the hero is anecdotally waining, the cultural aspect of the "lone engineer" or lone SME, working late nights to heroically finish the project does still exist in some organzations (again, acecdtoally). Another facet of this culture is the reliance on one or two senior engineers for all the design decisions – this can happen especially on teams with few seasoned engineers and many new ones (< 3 years experience). Engineering culture also fosters a tendency to procrastinate. The tendency to reward the "hero" who comes through in the end is a barrier to teamwork and to of identifying and addressing concerns early in a program through team interaction, propoer systems engineering discipline, and sharing of information. Everyone lauds the firefighter – but the building inspector rarely experiences the same adulation, praise or prestige.

Commented [RW88]: Is this true? Our speculation? This feels sujective and judgmental, although I must admit I do procrastinate from time to time.

Possible Barrier: Team segmentation

Teams may segment, or form subgroups, along functional lines, because of differences in opinions, or differences in goals. Some of this can be healthy, as all the structural engineers, co-located, can discuss projects and share expertise. In any large project, the decomposition of work into subprojects or disciplines is required to realistically achieve the project's goals. However, whenever a team forms subgroups, information flow could be impacted, and care must be taken to prevent these subgroups from undesirable divergence. This is where a strong lead systems engineer can help. Additionally, the ability to openly discuss and debate interactions and alternatives might be hampered by allegiances to the subgroup. Functional alignment was the most commonly sighted reason for teams forming subgroups. The resulting 'turf protecting' results in missed opportunities to leverage cross-domain knowledge. Active and savvy leadership – technical and line – can create a safe environment that minimizes the downsides of segmentation.

Commented [RW89]: Is this true, and if so what makes us think this? Reference?

Now that the foundation has been established, this paper moves onto the specific recommended phases to move to a CSTC.

Commented [RW90]: This was per Leema's comment, that CST has a few different meanings.

The Phases to Move to a CSTC

To cultivate a CSTC culture per reference [1] it is needed an environment that values system thinking and transfer knowledge between team members. For best results, to introduce CSTC into a corporation it is suggested to implement in phases. While there are other models for personal change, the list below is the recommendation specifically for CSTC in organizations. Table 1 lists a summary. The previous recommendation, to choose a core team of staff and management to lead the change, is essential and will be assumed.

Commented [ND91]: This was an idea (phases), but another name could be used.

Table 1. Phases for Implementing CSTC in an Organization

No.	Phase	Description

1	Awareness and Documentation	Collect situational awareness, employee interviews, identify pain points (that a CSTC could solve).
2	Investigation of Current State	Identify existing elements of a CSTC
3	Early Adoption and Management Buy-In	Here some small pilot projects are started, and management is approached for their support.
4	Methodology	The process for implementing the CSTC; very organization dependent and needs to be architected by leadership or the implementation core-team.
5	Removal of Barriers	Many may show up; keeping momentum is key at this phase until a tipping point in the organization can be reached.
6	Fill the Gaps	The CSTC rollout will expose gaps and holes in the new process – and maybe even the rollout itself. This is normal, and the gaps should be worked.
7	Training and Continuous Improvement	This sustains the gains and the victory, both with revised training and corporate policy if needed – and with continuous improvement efforts to keep the spirit of CSTC going and evolving.

- ***Phase 1 Awareness and Documentation*:**
  The first phase of the implementation of the CSTC is awareness and documentation. This involves understanding the culture of the corporation by conducting interviews and/or employee research on how *they perceive* the organization is run. How the employees perceive the vision and mission of the corporation may not align with the goals of the VPs, CEO. Identifying disconnects is important in the early phases.

  At the end of phase 1 the following questions should have an answer:
  - What is the mission and/or vision as defined by the corporation?
  - Do the employees have a shared understanding of this vision?
    - If not, how is the vision interpreted by the employees?
  - Does the current vision have a system thinking mentality?
  - What are the explicit and implicit incentives that drive individuals and organizations?
    - Do any of these contradict or impede achievement of the organization's mission or vision?
    - Do any of these contradict or impede establishment of a more collaborative systems thinking culture?
  - Are company practices and policies in alignment with the vision?
    - Are they in line with a system thinking culture?
    - If not, do employees have suggestions on how practices can be improved?
  - What are the main staff and front line management pain points?
    - Do the employees feel the organization is capable of fixing these?
    - Do any of the pain points, seem like a CSTC would solve them?
    - Are the employees, in a way asking for more collaboration, more systems-thinking without naming those terms? If so, this will help in later stages.

Commented [ND92]: are we using company or corporation?

- ***Phase 2 Investigation of Current State***:
  The second phase of the implementation is the investigation. Investigate if a CSTC, or parts of it, is practiced in the corporation. While the term system thinking culture may not be widely used, the employees of the corporation could have been practicing it without knowing it. This would be a huge step forward, as any change effort should leverage what already exists that can help. Then, additional concepts of the system thinking culture could be introduced as the need is identified. The investigation should include a variety of individuals from as many different groups within the company as possible in order to get a thorough understanding of interaction between various groups. Additionally, some basic training on systems thinking for those providing input to the investigate may help discover in unveiling existing systems thinking processes as well as potential areas of improvement. Training a small group of early adaptors will also assist with the next phase. In this phase, any informal leadership structures should be identified.

- ***Phase 3 Early Adpotion and Management Buy-in***:
  The third phase of the implementation is the elicitation of buy-in. This is suggested to consist of two parts: a 'bottom-up' piece that contains a demo project or small well-contained sample effort where a systems-thinking approach is used, and collaboration is emphasized. This small project, when successful, would then be documented as to how the new process, led to the success. Many change models suggest the early collection of 'small wins'.

  Second, management can be approached with this small victory as a lead in for future support. Obtain buy-in from the management of the corporation to implement the system thinking culture if not already in place. At this point, it is suggested to build on the CSTC elements that already exist, and cast the change effort as building onto what already works well. Buy-in by leadership is possibly the most critical phase for any change initiative., especially one that can change how an organization does business so profoundly like CSTC. Change is difficult and can often be resisted, therefore, it is important to focus on how a system thinking culture will not only benefit the company, but also the individual – hence the pain-point collection in Phase 1. Note that buy-in follows a progressive model, and not everyone will jump on board at once. The following are potential benefits that both management and senior staff may respond to, through the phase of getting company buy-in.

  - System thinking culture may require a change in how things are done, and take an investment to implement, but will provide ample opportunities for identifying improvements. In fact, the CSTC nature will find improvements and spot gaps more quickly, leading to a faster time-to-fix in the organization.
  - System thinking helps to ensure the work of an individual is streamlined by providing them with the information and tools that benefit their work (ie required information is readily available before it is needed vice only after being requested). Again this translates into speed.
  - A CSTC empowers individuals to make changes without fear of treading in someone else's lane. The communication paths are already established to prevent unwelcome intrusions. The culture paradigm requires issues and problems be voiced and worked towards improvement. Management does not have to be the font of all improvement work.
  - The fruits of the demo project should be mentioned: benefits and problems discovered during the previous phases should provide specific examples of what a culture change may improve for both company and individual. Some hint or evidence of an ROI would help here.

Here and in the later phases, employees should be given a voice on the development of the methodology and implementation.

- ***Phase 4 Methodology:***
  The 4th phase of the implementation of a system thinking concept is the methodology and implementation plan to introduce the CSTC. At the beginning of this phase, it is essential to establish a clear set of goals that will help to achieve your mission, but should also have buy-in and reflect the input of the team/employees that will be affected by them. Continuous communication allows team members to develop ideas for process improvements and discuss them to get them approved, funded and implemented. As the methodology is developed, everything should be compared to the goals and to the system thinking ideology.

  This is very organization dependent, but the methodology will likely include steps such as:
  1. Establish and require certain training, including on any new tools being used.
  2. Model the demo project on some chosen larger ones. Do not choose the highest-visible, or most expensive project at first.
  3. Assess and monitor CSTC progress, and also staff morale and acceptance.
  4. Prepare any communications to the sponsoring executive (one should be found, and act as the 'champion').
  5. If deploying this on an active project, then establish any expectation management with the customer if the new project plan looks very different from the old one.
  6. Identify and create any new engineering procedures, planning guidance, and even project cost/schedule estimation tools, to reflect use of these CSTC procedures and processes. While policy rarely drives behavior without a lot of "force majeure" – once the new behavior appears, it is recommended to backfill policy as appropriate to support the new behaviors.
  7. Continue until the pre-defined 'endpoint' of either activity or results are reached in the implementation effort. Measuring the affects of culture can be hard, but for the sake of employee morale it is recommened to declare some type of endpoint to the push.

  Techniques and ideas from Agile can be used to promote the communication and collaboration with team members. Additionally, the methodology should include a plan for positive conflict resolution in order to better enable the next phase, and some select metrics or measuring sticks to to determine if the effort has been successful. For instance – are projects coming in on time at a higher rate?

- ***Phase 5 Removal of Barriers:***
  The 5th phase of the implementation is the removal of barriers. In growing a CSTC the removal of barriers may include reducing the multiple hierarchy levels and approval layers to perform a task. This should result in a more efficient team and may be the greatest source of quick wins. Other ideas include (but are not limited to)

  - Rearranging the furniture to have space for group meetings to quickly discuss ideas and projects promotes collaboration.
  - Remove silos by integrating different groups disciplines in key discussion and decisions.
  - Perform peer reviews and working groups that include those different groups.
  - Identify, and try to work with those portions of the organization that does not like this change. This group will exist.

Commented [RW93]: Does anyone know of the reference

It is critical for leadership to be encouraging in the removal of barriers so as not to (unintentionally) be a barrier themselves. This phase is likely to cause "storming" (from the forming/storming/norming/performing model) within the team as ideas of what is and is not a barrier will be different to everyone. It is important to keep the team open to new and possibly-strange ideas. This is also a place where tailoring to the needs of the organization can take place. Some barriers like safety checks, are there for a reason. A best practice for barrier removal is to enable "trial periods" were the change initiated is given enough time to be evaluated before deciding if it should be tailored, discontinued, or if no change is needed and progress should continue. At this point, the commitment of leadership may be tested – as the natural resistance to change will have appeared by this time. Only leadership can motivate (or move? remove?) the stragglers.

- ***Phase 6 Fill the Gaps:***
  The 6th phase of the implementation is filling the gaps. Gaps differ from barriers in that there is something missing that is needed which would improve a team's ability to get the work done. These Gaps may be in knowledge, or in tools or even in adequate workspace. The gap could be a new team would have to be stood up, for instance a Requirements Task Team (for interfaces). Some gaps could be solved by creating Communities of Practice or Lunch & Learns for information exchange and training. These types of groups help to fill any gaps in knowledge from System Engineering participants. In these forums participants can showcase work, research and lessons learned. Additionally, a periodic status meeting should be held between the team and leadership to communicate findings and progress. These meetings should also be used to assess the ongoing efforts and their alignment with the goals developed in phase 4. The learning along the way, will expose holes in the process; these should be worked to keep the momentum and facilitate the benefits listed at the beginning of this paper.

- ***Phase 7 Training and Continuous Improvement:***
  The 7th phase of the implementation is the training of the new employees, current employees, and management on what the new processes and procedures are. This will be a thinly veiled version of what a successful system thinking culture should be for that organization. The training should explain the reason why a change on culture was needed, its impact, and stress that the cultural aspect was as important as the technical ones. At this point it is needed to establish common Systems Engineering terminology and concepts – some would say the new ontology needs to be established. This would facilitate its adoption, and reduce confustion on the new approach to engineering work. The new way should not be more confusing than the old one! The vision is that the use of collaborative systems thinking in the team, and the larger group, becomes the new normal, establishing it as part of the corporate culture.

At this point, the intent is that even with an imperfect rollout, and obstacles to some of the CSTC goals – there is consensus on the value of the effort among staff and management. This looks like less rework, smoother communications among teams, and fewer management surprises for much of an organization's project work. Additional benefits from a CSTC include the early feedback in the design and with the concurrence of the customer, This makes early design prototypes, either in hardware or in a modelling tool closer to what the customer will want; this reduces the design cycles. This approach helps establish good relationships with customers while completing the project on budget and as scheduled, because with the benefits of a CST culture rework is reduced With systems engineering becoming more collaborative within the organization, it becomes more customer-centric

as well. The "engineering" work expands into more than the technical realm – customer relations, program management, communications planning. Intangibles like trust, collaboration, customer satisfaction, even quality become part of the systems engineering-mix. This process is of course tailorable based on the organization's need and appetite for (tolerance to) process improvement. A culture change or shift is not an easy task, but sometimes it is necessary to adapt to a changing environment and the needs of the customers. An organization that embraces completely CSTC is not afraid to to listen to the customers and their employees. This courage leads to the long term health and sustainability of the organization, and its ability to win business now and in the future.

Conclusion

Commented [LJ94]: @Raymond Wolfgang

This paper has covered the concepts of collaboration, systems thinking, and how they can (and should) interact in a systems engineering environment. The need for a collaborative approach along with and combined with a systems-mindset and systems-thinking mentality was presented, along with a process to help the reader bring about such a transition in their organization. In presenting this we return to the question, of why would we pursue a Collaborative Systems-Thinking Culture in the first place? Can't the old methods, eventually, solve complex problems also? What about all the great engineering in the 60's and 70's, which was done before a lot of systems theory was developed? While not entirely true [ref: The Secret of the Apollo, Johns Hopkins Press], but is a stereotype found in some organizations that have been around for decades. Fundamentally, the reason is to save both time and money in developing technical products and services. Budgets are not expanding, and customers across industries consistently push for "more for less".

Commented [MB95]: You can quote few authors about how the application of 70's SE was not suitable for social systems https://www.linkedin.com/pulse/apollo-11-mazzucatos-mission-economy-critical-systems-jackson-obe/?trackingId=6VNwDswhKKSlPxreq753MA%3D%3D

If we can adopt not just practices – but the *mindsets* – needed to reduce rework, cut development time, increase customer satisfaction as well as boost our organization's overall contribution to society then we are obligated to act in this direction. In fact, the current economic and global business environment forces us as organizations to adapt in this direction or become defunct – or worse, irrelevant and in need of subsidy and assistance to stay afloat. It is not just our organizational solvency that is at stake – but lives. Examples abound – from the logistical effort seen to distribute the Covid-19 vaccine – to the recent tragedies in air travel that have been traced back to failures in the systems engineering (specifically, the human factors aspect) - adopting a systems-thinking mindset and culture has become even more pressing. This says nothing about the potential of engineering to solve some of the most vexing problems of humankind – fresh water, food distribution, disease prevention and control among others. We as engineers and leaders have an opportunity to contribute – and contribute substantially; leading the way to a Collaborative Systems-Thinking Culture in ourselves and our organizations is the brick-and-mortar to paving this new path.

A large bibliography describes the future challenges of the system engineering – including INCOSE's own forthcoming Vision 2035. Organizations will have to maintain business with disruptions in politics, the environment, social systems, and technology. Economic uncertainty and continuing struggles with education and the legal environment will add to the complexity. This is just what we know about and can predict in 2022! The resulting challenges will require the transforming potential of systems engineering to better integrate business, engineering, problem-solving in a cultural context that shows no sign of stabilizing or becoming predictable. With a Collaborative System Thinking Culture, both at an organizational level and even adopted as an individual mindset, a system engineer will be armed to address coming new forms of complexity and grow in his position of leadership in engineering domains. It is not new, that systems engineers have expanded beyond purely technical pursuits; CSTC has led us in the right direction to incorporate non-technical skills as part of our collaborative mindset. Engineers now must expand on systems thinking to take on the larger challenges societies face – this not a moment too soon, and our work as systems engineers is not

nearly done. The CSTC approach allows us as systems engineers to progress in this important work. We are now leaders, ready to contribute.

Commented [MB96]: Can make it much longer but paper is 19 page which is already too long.

Reviewers: Jay Patel, @Jason Baker

Commented [LJ97]: @Jason Baker

References

Commented [AF98]: References shall be reviewed and ordered

1. Lamb, C.T and Rhodes, D.H., "Collaborative Systems Thinking: Uncovering the Rules", Massachusetts Institute of Technology, November 2010, IEEE A&E Systems Magazine v25 n 11.
2. Lamb, C.T and Rhodes, D.H., "System Thinking as an Emergent Team Property: On going Research into Enablers and Barriers to Team-Level System Thinking", IEEE International Systems Conference, IEEE, New York, 2008.
3. Rodenas M. A., Jackson M. C., "Lessons for systems engineering from the Segura River reclamation project: a critical systems thinking analysis", Wiley, 2021
4. http://www2.csudh.edu/gsterling/complex.pdf
5. Bourdieu, P. (1986). "The Forms of Capital. Readings in Economic Sociology" J. Richardson (Ed.), "Handbook of Theory and Research for the Sociology of Education, vols. 241–258", Greenwood Press, New York.
6. Snowden, D., Greenberg, R., "Cynefin - Weaving Sense-Making into the Fabric of Our World", Cognitive Edge - The Cynefin Co., 2020
7. Cognitive Edge - The Cynefin Co., 2021, https://www.cognitive-edge.com/the-cynefin-framework/
8. Brougham, G., "The Cynefin MiniBook", Lulu.com, 2015
9. Boardman, J. and Sauser, B., *Systems Thinking: Coping with 21st Century Problems*, CRC Press, 2008
10. Boehm, Barry, 2011. "Some future software engineering opportunities and challenges." In *The future of software engineering*, pp. 1-32. Springer, Berlin, Heidelberg.

Commented [AF99]: Example…