SYSTEMS ENGINEERING
Research Center

Atlas: The Theory of Effective Systems Engineers, version 1.0

December 16, 2016

Research Team:

Dr. Nicole Hutchison, Dr. Devanandham Henry, Dr. Dinesh Verma
Megan Clifford

Stevens Institute of Technology

Dr. Art Pyster

George Mason University

Work conducted under Research Topic 154

Contract No. HQ0034-13-D-0004 Task Order 0054, RT 154

The Systems Engineering Research Center (SERC) is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology.

This material is based upon work supported, in whole or in part, by the U.S. Department of Defense through the Assistant Secretary of Defense for Research and Engineering (ASD(R&E)) under Contracts H98230-08-D-0171 and HQ0034-13-D-0004.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

The Helix team would like to thank all the organizations and individuals that willingly participated in the project, offering their resources, time, and effort. This was critical to our research. Their active participation in the Helix interviews provided us data that was rich in both quality and quantity, which makes this research more valuable and useful to the participating organizations and the systems engineering community at large.

We are most grateful to the Office of the Deputy Assistant Secretary of Defense for Systems Engineering (DASD(SE)), especially Kristen Baldwin and Scott Lucero, for their continued support, without which this research would not be possible. The International Council on Systems Engineering (INCOSE) and the National Defense Industrial Association Systems Engineering Division (NDIA-SED) are both valued partners in this research and we thank them, especially Courtney Wright, David Long, Bill Miller, and Don Gelosh.

We also thank all former members of the Helix research team whose contributions have shaped our research over the years. In particular, we would like to thank Dr. Art Pyster, whose leadership and dedication since the beginning of the project were critical and instrumental to its success.

Nicole AC Hutchison

Helix Principal Investigator

Executive Summary

Atlas 1.0 is the culmination of over four years of research into what makes systems engineers effective. The key elements that play a role in effectiveness are identified in Figure 1 below. The specifics defined for each of these variables are the result of in-depth research on systems engineers. Additionally, related disciplines such as classic engineering (electrical, mechanical, software, etc.) or systems-related professions are also expected to find these materials applicable with slight tailoring.

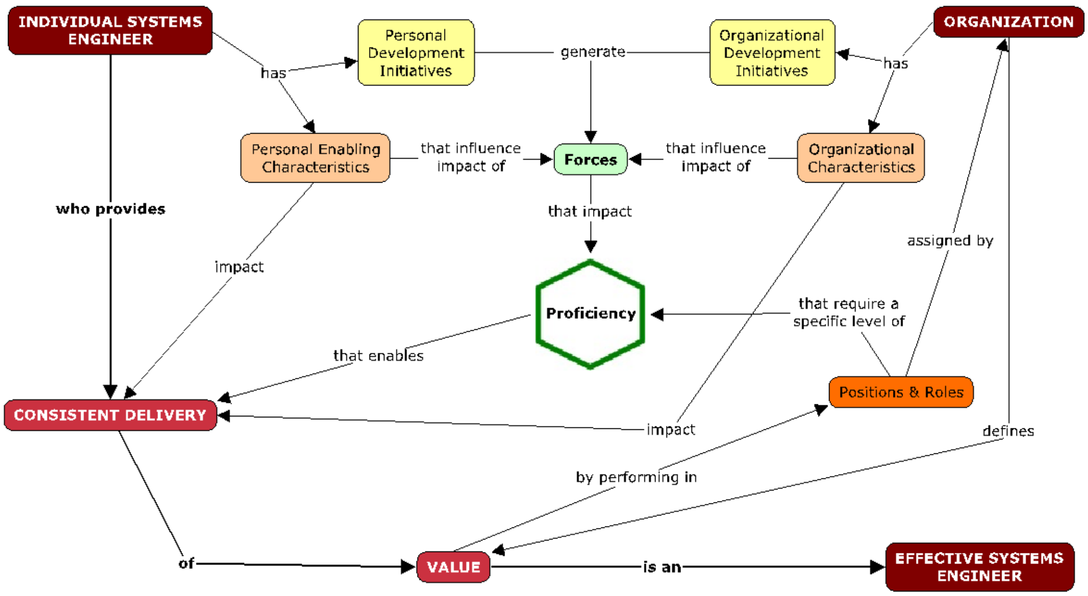


Figure 1. *Atlas 1.0* Overview

The main theme of *Atlas* is an *Individual Systems Engineer who provides Consistent Delivery of Value is an Effective Systems Engineer*. This definition hinges on Value, which is defined by the Organization in which a systems engineer is working. Value is created by working in defined positions and roles. The organization must establish the position of the systems engineer in terms of roles and responsibilities and this should align with specific levels of Proficiency – knowledge, skills, abilities – that enable a systems engineer to perform in a given position.

Both individuals and organizations may have development Initiatives; together, they generate forces – experiences, mentoring, or education and training – that impact proficiency. At the same time, personal and organizational characteristics influence the impact of forces on proficiency – positively or negatively. Both personal and organizational characteristics impact consistent delivery of value. Amidst these variables and their interactions, the challenge for the individual systems engineer and the organization is to improve the proficiency that enables consistent delivery of value to the organization.

Atlas is expected to be used in several ways: first, by individuals who wish to better understand their own proficiencies and effectiveness in the context of their organization; second, by organizations that wish to understand the current state of the effectiveness of their systems engineers; and third, by either individuals or organizations for future career planning. These use cases and recommended approaches are described within this document.

Additional details regarding methodology and background on the development of *Atlas 1.0* can be found in the companion Technical Report (SERC-2016-TR-118).

1 BACKGROUND AND INTRODUCTION

The Systems Engineering Research Center (SERC), a University Affiliated Research Center (UARC), set up by the U.S. Department of Defense (DoD), responded to the systems engineering workforce challenges by initiating the Helix Project to investigate the “DNA” of systems engineers, beginning with those who work in defense and then more broadly. The US Deputy Assistant Secretary of Defense for Systems Engineering (DASD(SE)), the International Council on Systems Engineering (INCOSE) and the Systems Engineering Division of the National Defense Industrial Association (NDIA-SED) jointly sponsor Helix. To ensure Helix delivers the greatest value and to help Helix obtain access to the necessary data, Helix formed the Helix Advisory Panel (HAP) with representatives primarily from those three sponsor organizations. Helix has held three annual workshops with a broad set of representatives from across government, academia, and industry.

Helix is a multi-year longitudinal research project, which has gathered data from many organizations with DoD and the Defense Industrial Base (DIB) through a combination of techniques, including interviews with hundreds of systems engineers. In 2014, Helix began to reach beyond DoD and the DIB, to gather data from other types of organizations as well, including non-defense organizations in the US and non-US organizations. Version 0.25 of *Atlas* was also published in 2014. *Atlas* identifies the key variables that impact a systems engineer’s effectiveness – positively or negatively – and provides, as much as possible, details on how these variables impact effectiveness.

During 2015, Helix expanded its data collection by conducting interviews with non-DoD organizations as well; matured *Atlas* into the next version, *Atlas 0.5*; defined and analyzed the career paths of systems engineers; and did implementation trials of *Atlas*.

During 2016, the team generated *Atlas 0.6* and *Atlas 1.0*. *Atlas 1.0* reflects the results of analysis of in-depth interviews with 287 individuals. Most of these individuals were systems engineers, though approximately 10% of the sample was comprised of individuals who work with systems engineers – organizational leaders, classic engineers (electrical, mechanical, software, etc.), and program managers. In 2016 the Helix team also worked on implementation of *Atlas* with a number of organizations and lessons learned from those activities are captured here.

1.1 HOW IS *ATLAS* DIFFERENT FROM HELIX?

Helix is the name of the overarching SERC project. Helix has been examining what makes systems engineers effective for over four years. As a project, Helix has created many different deliverables or products. The primary product of Helix is *Atlas: The Theory of Effective Systems Engineers*. This document represents *Atlas 1.0* – expected to be mature enough for individuals or organizations to use without direct help from the Helix team. It is a standalone document to detail the contents of *Atlas*.

This document does *not* contain all of the research that led to the development of *Atlas 1.0*. Instead, the detailed research results and how they led to *Atlas 1.0* are contained in the companion Helix Technical Report (SERC-2016-TR-118). Individuals or organizations that want not just to use *Atlas* but to also understand the rationale and methodology behind its development should reference the Technical Report. Several earlier published Helix papers and technical reports are also referred to throughout this report. The reader is not expected to read the earlier technical reports or any of the other Helix papers or reports, in order to understand *Atlas 1.0*.

In addition, there are tools that an individual or organization can use to support self-assessment using

Atlas. The paper-based tools are contained in the Appendices of this report. The team has also developed more easily tailored Excel-based tools, which can be found on the Helix page of the SERC website (http://www.sercuarc.org/projects/helix/).

The relationship between Helix, *Atlas*, the Technical Reports, and the tools is illustrated in Figure 2.

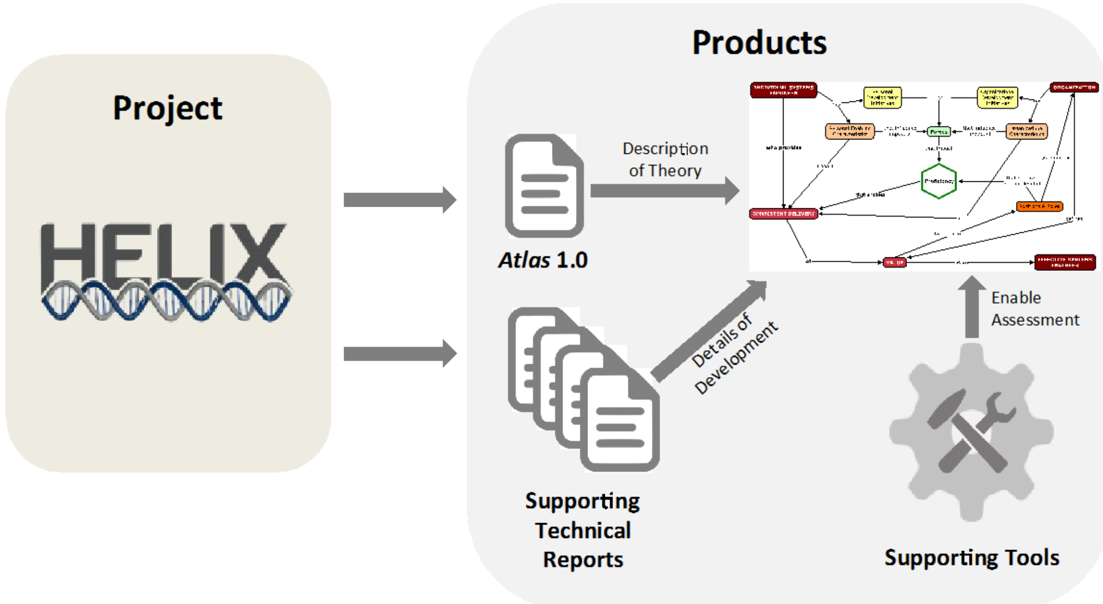


Figure 2. Relationship between Helix and *Atlas*

1.2 Incremental *Atlas* Development

The Helix project used an incremental approach to develop *Atlas*. This approach was designed to enable publication and use of aspects of *Atlas* as they became appropriately mature, while maintaining the expectation that *Atlas* would become more mature over time. The increments were:

- ***Atlas 0.25***: The first draft of *Atlas* based on work done in 2014 was published as *Atlas 0.25* in November 2014. It included key elements that explain the effectiveness of systems engineers, and a preliminary explanation of the relationships between those elements. The structure and variables of the proficiency model were also included, along with some initial analysis of career paths.
- ***Atlas 0.5***: Based on subsequent work done in 2015, *Atlas 0.5* was published in December 2015. It reflected further understanding of the elements of *Atlas* and their inter-relationships. Significant new work was done in the area of career paths and *0.5* incorporated initial efforts to use *Atlas* to assess the level of proficiency of systems engineers. *Atlas 0.5* was mature enough for an individual or an organization to use and gain valuable insights with some guidance from the Helix team.
- ***Atlas 0.6***: Was an incremental improvement to *Atlas 0.5*. It contained additional detail and analysis for areas that were less mature in 0.5, namely: mentoring, personal initiatives, and organizational initiatives. *Atlas 0.6* was not created as a stand-alone document, but rather as a supplement to 0.5.

- ***Atlas 1.0***: *Atlas 1.0* – this document – includes a more complete description of the elements of *Atlas* and their inter-relationships. *Atlas 1.0* is believed to be mature enough for independent deployment and assessment by individuals and organizations with little or no guidance from the Helix team. In addition, the frameworks presented in *Atlas 1.0* have been validated using data from outside the US DoD, and therefore is believed to be applicable to systems engineers in a variety of domains. This is intentional. Though the initial impetus for the work was based on the needs of the US DoD, the Helix team believes that a more generic framework which benefits all systems engineers, regardless of domain, is both more beneficial to the community at large and, ultimately will benefit the US DoD by setting consistent expectations for practitioners across domains.

Atlas 0.25 and *Atlas 0.5* were mature enough for trial. The Helix team is aware of five organizations that have used some aspects of *Atlas*, primarily to assess the proficiency levels and understand the career paths of individual systems engineers within the organization. Feedback and observations from these early use exercises influenced the development of *Atlas 1.0* as published here. A glimpse into potential benefits of *Atlas* deployment, based on trials conducted in 2015 and early 2016, are included throughout this report, with findings related to each element of *Atlas* reflected in corresponding sections on that element.

1.3 About This Document

This document reflects *Atlas 1.0: The Theory of What Makes Systems Engineers Effective*. This document:

- Provides an overview of *Atlas 1.0* (Section 2);
- Provides details on the elements of *Atlas 1.0* (Sections 3-8);
- Provides insights on how these elements come together over time into a Career Path (Section 9);
- Provides insight on the expected use cases for *Atlas 1.0* and how *Atlas* is expected to be tailored for implementation (Section 10); and
- Provides the paper-based tools that can be used for assessment. (Appendix A)

In addition, within each section considerations for use of *Atlas* by individuals and organizations are provided. With these materials, the Helix team believes that any individual or organization can begin utilizing *Atlas* without direct support from the Helix team. However, the team would be glad to receive feedback and to address any issues, concerns, or questions from the community and can be contacted at helix@stevens.edu.

2 INTRODUCTION TO *ATLAS 1.0*

Atlas is a set of general principles and ideas that relates to the subject of what makes systems engineers effective and why. In doing so, *Atlas* also provides insights into how individuals can develop into effective systems engineers throughout their careers and what organizations can do to support this development.

2.1 *ATLAS* OVERVIEW

The overview of *Atlas* in the context of an individual systems engineer employed in an organization is captured in the systemigram illustrated in Figure 3. A systemigram consists of nodes that contain noun phrases, links that contain verb phrases, and is to be read as sentences along the direction of the arrows. The primary sentence is read from the top left node to the bottom right node and presents the main theme of the systemigram. In the ensuing discussions, sentences to be read in the systemigram are italicized, where nodes are represented in square brackets.

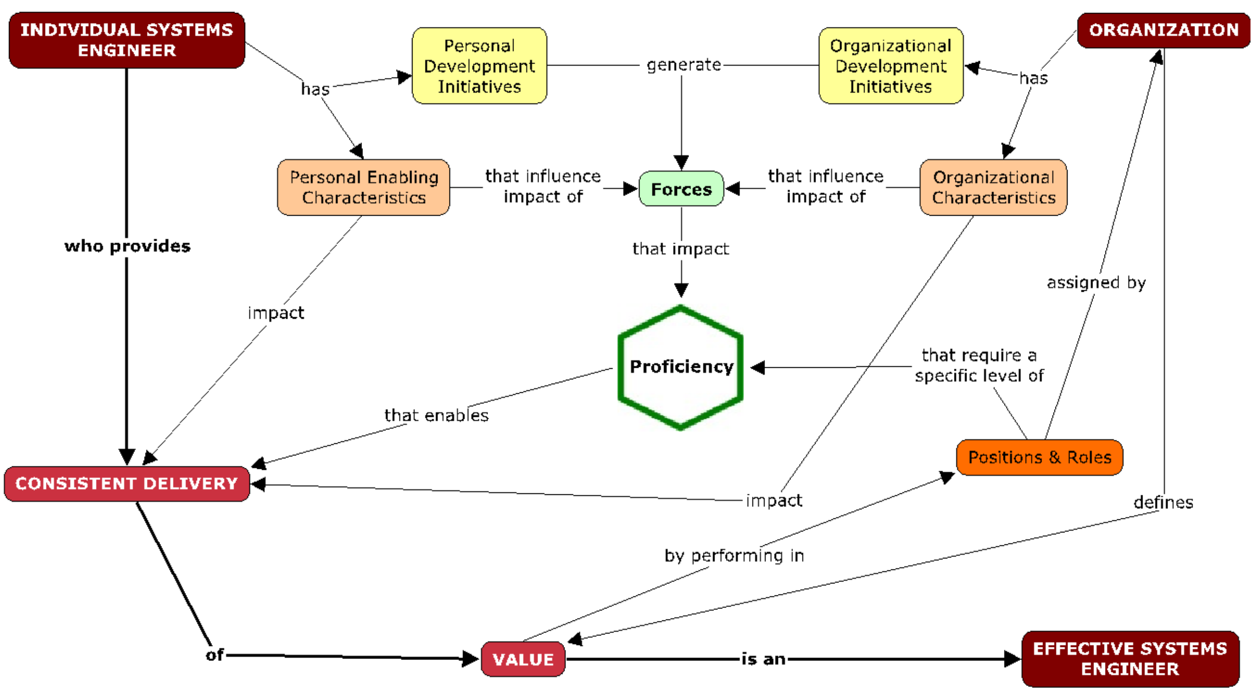


Figure 3. *Atlas 1.0*

From Figure 3 above, it can be seen that the main theme of *Atlas* is: '*[Individual Systems Engineer] who provides [Consistent Delivery] of [Value] is an [Effective Systems Engineer]*'. This fundamental definition of an effective systems engineer hinges on *[Value]*, and it can be seen that '*[Organization] defines [Value]*'. Therefore, it is on the organization to define the value that the systems engineer is expected to provide. Further, the individual systems engineer provides '*[Value] by performing in [Positions and Roles] assigned by [Organization]*'. Therefore, it is again on the organization to establish the position of the systems engineer in terms of roles and responsibilities, keeping in mind that '*[Positions and Roles] require a specific level of [Proficiency] that enables [Consistent Delivery] of [Value]*'.

The core of *Atlas* is the proficiency of the individual systems engineer – what proficiency means, and how it can be improved. '*[Individual Systems Engineer] has [Personal Development Initiatives]*' and '*[Organization] has [Organizational Development Initiatives]*'; together, they '*generate [Forces] that impact [Proficiency]*'. At the same time, '*[Individual Systems Engineer] has [Personal Characteristics] that influence the impact of [Forces]' and '[Organization] has [Organizational Characteristics] that influence the impact of [Forces]'* – these forces may have a positive or a negative influence. Further, both personal enabling characteristics and organizational characteristics '*impact [Consistent Delivery] of [Value]*'; again, the impact can be positive or negative. Amidst all these influences and impacts, the challenge for the individual systems engineer and the organization is to improve the '*[Proficiency] that enables [Consistent Delivery] of [Value]*' to the organization.

The color-coding of the systemigram (Figure 1) is designed to show the relationships between various elements of *Atlas* as follows:

- The mainstay – the primary focus of the systemigram (Boardman and Sauser 2008) – is in red.
- Primary actors are in dark red (individual systems engineer and organization, leading to the desired end state of effective systems engineer).
- Elements related to the skills of systems engineers – the specific skills themselves or how they are developed – are in green.
- Characteristics of the primary actors are in orange.
    - "Positions and Roles" is called out as a darker orange because it is related to both the individual and the organization. Positions and roles are a characteristic of the current state of the individual and the mechanism by which she would deliver value. These are also determined by the organization, and the way an organization defines positions is related to other organizational characteristics.
- Initiatives of the primary actors are in yellow.

Note that the terminology used in Atlas will be defined in relevant sections via call-out boxes.

2.2 Dynamic Aspect of *Atlas*

The *Atlas* overview illustrated in Figure 3 can be considered as a quasi-static snapshot in time, but many of the elements of *Atlas* are dynamic in nature. The level of proficiency of an individual systems engineer is not fixed, but is constantly changing due to the impact of forces over time. Similarly, other elements of *Atlas*, including characteristics and initiatives of the individual systems engineer and of the organization, continue to change over time. Further, as the level of proficiency of an individual systems engineer increases over time, the organization is likely to place that systems engineer into different positions.

This dynamic aspect of *Atlas* is not captured in the overview, but is reflected in the career paths of individuals over time, where an individual's career path is the precise combination of the forces they undergo in the positions and roles they perform in over their entire career.

Leading up to the publication of *Atlas 1.0*, the Helix team defined methods to depict and analyze the career paths of systems engineers and used those methods to analyze the systems engineers in its interview sample, and how those systems engineers are shaped by the impact of forces and positions and roles over time.

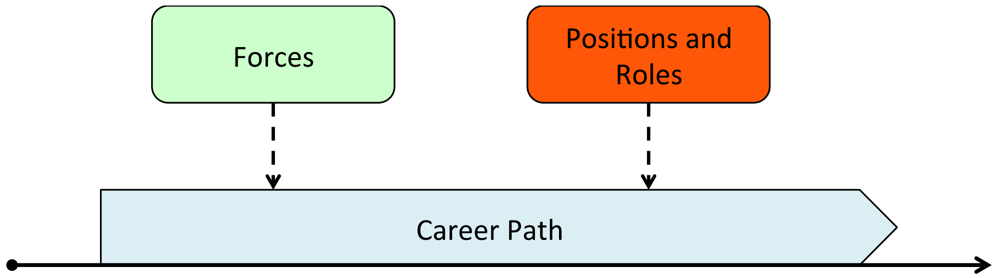


Figure 4. Career Path: A Dynamic View of *Atlas*

The Helix team has defined methods to depict and analyze the career paths of systems engineers. The team used those methods to analyze the systems engineers in its interview sample and to understand how those systems engineers are shaped by the impact of forces and positions & roles over time.

3 VALUE OF SYSTEMS ENGINEERS

The broad question that Helix is trying to address is: 'How to develop effective systems engineers?' The key term in this question, in addition to a consistent understanding of who is a systems engineer, is '**effective**'. When initially asked who an 'effective systems engineer' was, interviewees tended to give the response '*one who develops (or supports development of) systems within time, cost, and schedule constraints*'. This definition was not very insightful, and hence Helix developed an alternative definition – an effective systems engineer is '*someone who consistently delivers value by performing systems engineering activities*'. This definition introduced the term 'value', and thus provided a context for effectiveness. Of course, value by itself is a subjective term, and was not something that Helix wanted to define up front. Instead, Helix wanted to understand what systems engineers said was the value they provided and to understand what non-systems engineers said was the value that systems engineers provided.

effectiveness – the ability to consistently deliver value.

systems engineer – an individual who performs systems engineering activities and is recognized (either formally or informally) by his or her organization for her ability to perform these activities.

effective systems engineer – someone who consistently delivers value by performing systems engineering activities in positions assigned by the organization.

value – the benefits gained through the application of systems engineering activities, as distinct from benefits gained through other disciplines.

The Helix team probed on the concept of value in 100% of the interviews conducted. The discussion of value took two general forms: an individual's perspective of the primary value that she provides as a systems engineer and an individual's perspective of the overarching value that systems engineers in her organization provide. Some individuals answered the value question in ways more readily linked with *proficiency* than value; for example, they might have referenced communication skills or deep understanding of systems engineering processes. As indicated above, a number of systems engineers also defined value in terms of overall project success ("on time, within budget"), which does not allow specific insights for systems engineers versus project managers or any other personnel who support the project. After filtering these types of responses, there were 313 individual excerpts on the value that systems engineers provide offered from 85 individual systems engineers.

The key values identified are provided in the list below. The main bullets state the overarching values that systems engineers provide; the sub-bullets are the ways these values are achieved, often discussed as enabling or lower-order values. Percentages reflect the percent of the data related to a given value or the relationship between values. So for example, the first value, "Keeping and maintaining the system vision", was described in 11% of the excerpts on value. However, in 39% of the areas where "Keeping and maintaining the system vision" was discussed, understanding of the customer's true requirements was described as a key enabling value. In some instances, percentages are not provided; these areas require additional analysis.

The primary values that systems engineers provide – as consistently stated across organizational and domain lines – include:

- Keeping and maintaining the system vision (11%) is enabled by:
    - Getting the "true" requirements from the customer and creating alignment between the customer and the project team. (39%)

    - Seeing relationships between the disciplines and helping team members understand and respect those relationships. (33%)
    - Balancing technical risks and opportunities with the desired end result. (36%)
    - Providing the big picture perspective for the system. (44%)
- Enabling diverse teams to successfully develop systems. (10%)
    - Effectively understanding and communicating the system vision to the team, and ensuring that the team is aligned with this vision. (38%)
    - Helping the team to understand the big picture perspective and where they fit within the larger picture. (38%)
    - Identifying areas of concern for integration in advance. (13%)
- Managing emergence in both the project and the system (7%)
    - Projecting into the future (14%), which includes staying "above the noise" of day-to-day development issues and identifying pitfalls.
    - Technical problem-solving balanced with the big picture perspective. (43%)
- Enabling good technical decisions at the system level (7%)
    - The ability to see the vision for the system and communicate that vision clearly is a key enabler to helping teams make good technical decisions. (40%)
    - The big picture perspective is critical for understanding the system holistically and enabling system-level technical decisions, versus decisions made at the component or sub-system level. (22%)
    - A systems engineer's solid grasp on the customer's needs is also a critical enabler to ensuring that decisions made will keep the system on the correct technical path. (22%)
    - Being able to bring together a diverse team of engineers and subject matter experts is also critically important. (26%)
    - A systems engineer's problem solving abilities – particularly the ability to focus on root versus proximal cause – is also a key enabler. (26%).
- Supporting the business cases for systems (7%)
    - Balancing traditional project management concerns of cost and schedule with technical requirements. (41%)
    - Understanding the position of a system within the organization or customer's portfolio and communicating this to the team. (59%)
- Translation of technical jargon into business or operational terms and vice versa (11%)
    - Translating highly technical information from subject matter experts into common language that other stakeholders can understand.
    - Translating operational concepts, customer needs, and customer desires into language that makes sense for engineers and program managers who do not have the same understanding of the systems' future operating environment.

These values represent the combined perspective from all systems engineers across all organizations – a cross section of government and industry organizations from multiple domains. These were seen as the consistent values and no major differences were seen between government and industry or across different domains. However, it is worth noting that the means for delivering value was different.

For example, whether in the defense sector or other sectors, systems engineers in government organizations tended to be more focused on providing value by emphasizing standard processes, while commercial organizations tended to focus more on delivering the "right" end results by asking good questions, generating a vision for the system, and providing the big picture perspective. This ***does not*** mean that systems engineers in government organizations value process over the end result of systems development; instead, it means that in an acquisition environment – which was the context for the majority of government systems engineers – following a rigorous process was seen as a primary way to provide the values listed above and help achieve end results. In commercial companies, process was discussed, but not seen as the primary means for providing value. Systems engineers in commercial companies did state that systems engineers provide value by bringing a logical approach to problem solving and, in some organizations, processes were seen as a way to institutionalize these types of approaches, although with varying degrees of success. It is worth noting that systems engineers in commercial organizations in highly regulated industries tended to emphasize process more strongly than their counterparts in less-regulated industries.

3.1 Implications for Use

The values that systems engineers provide, outlined in *Atlas,* provide a good starting point for organizations to define their end-state expectations for systems engineers' effectiveness. This does not mean that every organization will expect every value from each systems engineer – and likewise there may be expectations from the organization that are not captured in *Atlas*. However, using this list and adding or editing as appropriate for the organizational context can help to set very clear expectations about what a successful systems engineer delivers. This information can then be communicated in a variety of ways: individuals will understand the goals and expectations more clearly; managers will have a clearer basis on which they are making personnel selections; and leadership will develop additional language to clarify who systems engineers are and what the expected benefits to systems engineers' participation in teams will be.

4 Positions and Roles

An individual systems engineer fills a *position* (or holds a title) in an organization, and there are many *roles* that the systems engineer is expected to perform in that position. *Atlas* identifies 17 systems engineering roles; typically, a systems engineer performs a combination of these roles while holding a single position. Starting with the 'twelve systems engineering roles' identified by Sheard (1996). The Helix team recombined, renamed, removed, and added roles to reflect the Helix data collected during interviews about the activities systems engineers perform in organizations today. This was socialized with the community through conference papers and presentations, the Helix workshops, and through early adopter activities with several organizations.

position – the particular arrangement of roles and responsibilities for an individual, as defined and assigned by the organization. Often, positions are equivalent to an individual's title.

role – a set of specific, related systems engineering activities.

4.1 Atlas Roles Framework

Tables 1-3 provide the roles of systems engineers and offers an explanation of how each role came to exist in the framework. For example, "System Integrator" is the role that was previously titled "Glue" in (Sheard 1996) and the name change as well as the rationale for the change is captured below. Tables 1-3 also highlight the *roles* framework developed, consisting of three categories:

- ***Roles Focused on the System Being Developed*** – These roles are what may most quickly come to mind when describing a systems engineer. They align closely with the systems engineering lifecycle and the critical activities systems engineers must enable throughout the lifecycle.
- ***Roles Focused on SE Process and Organization*** – These roles focus on the organizational context in which systems engineering works and the critical role of systems engineers in providing guidance on how systems engineering should be used.
- ***Roles Focused on Teams that Build Systems*** – Systems engineering does not occur in a vacuum and is, instead, an intensely social activity. The roles in this category focus on enabling diverse, multi-disciplinary teams to be successful.

The categories help distinguish between the major types of activities that systems engineers provide.

Table 1. Roles Focused on the Systems Being Developed

Role Name	Role Description
Concept Creator	Individual who holistically explores the problem or opportunity space and develops the overarching vision for a system(s) that can address this space. A major gap pointed out to the Helix team – particularly when working to implement the findings of Helix – has been that of the development of an overarching system vision. This is a critical first step in the systems lifecycle, and several organizations stated that they believed it needed to be separately called out. In addition, when looking to the future of what systems engineers need to do (e.g., INCOSE Vision 2025 (2015)), the focus on early engagement and setting

	the vision was deemed critical.
Requirements Owner	Individual who is responsible for translating customer requirements to system or sub-system requirements; or for developing the *functional* architecture. This is unchanged from (Sheard 1996).
System Architect	Individual who owns or is responsible for the architecture of the system. This is an update of Sheard's "System Designer" role (1996). There was concern both at community events and during later interviews that nowhere in the presented framework did the critical role of systems engineers in architecture come out clearly. Some also argued that "Design" gave the impression that this role focuses specifically on the details of systems design over architecture.
System Integrator	Individual who provides a holistic perspective of the system; this may be the 'technical conscience' or 'seeker of issues that fall in the cracks' – particularly, someone who is concerned with interfaces. Likewise, there was concern over the word "Glue", which many expressed was not clearly descriptive enough.
System Analyst	Individual who provides modeling or analysis support to system development activities, and helps to ensure that the system as designed meets he specification. This is unchanged from Sheard's roles (1996).
Detailed Designer	Individual who provides technical designs that match the system architecture; an individual contributor in any engineering discipline who provides part of the design for the overall system. This is an addition based on the Helix data. While systems engineers do not always get involved with detailed design, in smaller organizations or on smaller projects it is more common. Likewise, systems engineers who had played this role explained that it was critical in developing their own technical and domain expertise as well as in understanding the design approaches of classic engineers.
V&V Engineer	Individual who plans, conducts, or oversees verification and validation activities such as testing, demonstration, and simulation. This is unchanged from Sheard's roles (1996).
Support Engineer	Individual who performs the 'back end' of the systems lifecycle, who may operate the system, provide support during operation, provide guidance on maintenance, or help with disposal. This was previously titled "Logistics and Operations Engineer" in Sheard (1996). However, in interviews and at community events, the Helix team received feedback that using this title gave the impression that this role was limited and did not encompass the full spectrum of systems engineers' activities at system deployment or post-deployment. Likewise, in several organizations, "logistics" and "operations" were seen as separate disciplines from systems engineering, which caused some contention in discussions. The renaming of this category is intended to address these issues.

Table 2. Roles Focused on Process and Organization

Role Name	Role Description
Systems Engineering Champion	Individual who promotes the value of systems engineering to individuals outside of the SE community – to project managers, other engineers, or management. This may happen at the strategic level or could involve looking for areas where systems activities can provide a direct or immediate benefit on existing projects. Sheard recommended that a role such as this, labeled in her work as "Systems Engineering Evangelist", be added in (2000).
Process Engineer	Individual who defines and maintains the systems engineering processes as a whole and who also likely has direct ties into the business. This individual provides critical guidance on how systems engineering should be conducted within an organization context. This is unchanged from Sheard's roles (1996).

Table 3. Roles Focused on the Teams That Build Systems

Role Name	Role Description
Customer Interface	Individual who coordinates with the customer, particularly for ensuring that the customer understands critical technical detail and that a customer's desires are, in turn, communicated to the technical team. This is unchanged from Sheard's roles (1996).
Technical Manager	Individual who controls cost, schedule, and resources for the *technical* aspects of a system; often someone who works in coordination with an overall project or program manager. This is unchanged from Sheard's roles (1996).
Information Manager	Individual who is responsible for the flow of information during system development activities. This includes the systems management activities of configuration management, data management, or metrics. This is unchanged from Sheard's roles (1996).
Coordinator	Individual who brings together and brings to agreement a broad set of individuals or groups who help to resolve systems related issues. This is a critical aspect of the management of teams. This is unchanged from Sheard's roles (1996).
Instructor/Teacher	Individual who provides or oversees critical instruction on the systems engineering discipline, practices, processes, etc. This can include the development or delivery of training curriculum as well as academic instruction of formal university courses related to systems engineering. While any discipline could conceivably have an instructor role, this denotes a focus on systems and is a critical component in the development of an effective systems engineering workforce. This is an addition to the Sheard roles (1996)

The role of "Classified Ad" systems engineer, as defined by Sheard (1996) was dropped from this framework. "Classified Ad" was a placeholder role Sheard used to acknowledge the many job postings for "systems engineers" reflected IT network or computer specialists (e.g., network systems engineer, IT systems engineer, or Microsoft systems engineer). In the Helix sample, none of the systems engineers for whom roles data was collected played this role, either currently or in the past. In addition, when this role was presented at various community events (Helix workshops in 2014, 2015, and 2016; presentations on Helix at INCOSE (Lipizzi, 2015, Jauregui, 2016), there was a strong recommendation to remove it from the framework to highlight what systems engineers do and to draw a clear distinction from positions that may be titled "systems engineer" but which do not bear resemblance to the practice of systems engineering.

Tables 1-3 outline the *systems engineering* roles. However, there were a few roles that were commonly seen throughout the Helix data sample. These are roles that may frequently be played by systems engineers. These include:

- *Organizational/Functional Manager* - Individual who is responsible for the personnel management of systems engineers or other technical personnel in a *business* – not a project or program – setting.
- *Program/Project Manager* - Individual who is not *directly* responsible for the technical content of a program, but works closely with technical experts and other systems engineers while maintaining overall project cost and schedule.

These roles, while not systems engineering roles, are things that many systems engineers do throughout their careers and which may help systems engineers develop some critical skills. Figure 5 provides a simple Venn diagram showing, from the Helix data, the overlap between systems engineering roles and roles held by systems engineers.

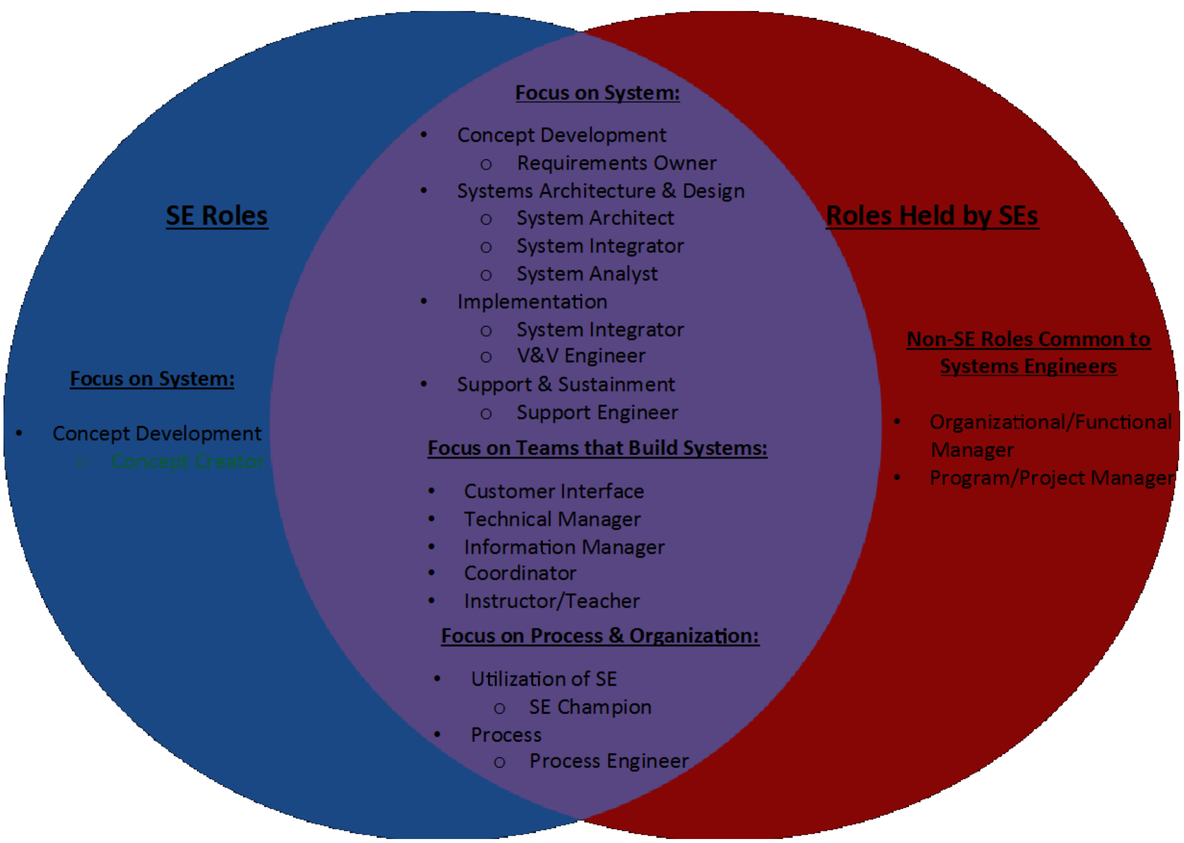


Figure 5.The overlap between SE roles and Roles Held by Systems Engineers in the Helix Sample

It may be surprising that one of the SE roles, "Concept Creator" (shown in green in Figure 5), is not a role that systems engineers in the Helix sample commonly played. A small number of individuals in the Helix sample did play these roles, but not enough, initially, to add this to the framework. The addition of this role was based on community feedback and work on implementation with several organizations. The Helix team believes that the primary reason that "Concept Creator" did not come out strongly in the sample is due to the organizations in which they work. In each of the government organizations that participated, systems engineers have been part of the acquisition workforce. When asked if they participated in initial concept definition, most explained that this was done before they were assigned to the system. Systems engineers at many industry organizations, particularly those within the DIB, expressed a similar view – that this early vision-setting happened before systems engineers got involved.

In looking to the future of systems engineers, there is a push for them to be included more in concept design. Clearly, concept development work is part of systems engineering as it is critical for successful systems, and one would assume that this would be an important role for systems engineers. This is reflected in strategic documents such as INCOSE *Vision 2025* (2014) as well as in the goals and desires of several organizations working to implement Helix findings and individual systems engineers. This is the rationale for inclusion in the *Atlas* roles framework.

4.2 Implications for Use

Individuals who are assessing their careers may benefit from the exercise of reviewing the roles which they did and did not play at various stages throughout their careers. (Templates in Appendix A.) This analysis will enable an individual to identify any clear patterns in their careers (e.g., a role that they have never played or a role that they have played consistently). This self-awareness about their own career will enable them to make more informed decisions about their future desired positions (e.g., identifying opportunities to play new roles).

There are several ways that roles assessments can benefit organizations. First, by enabling individuals to be more aware about their own careers, there may be some organic growth and development across the workforce. Second, understanding holistically any gaps or common patterns across the workforce may provide insight into potential Organizational Development Initiatives. For example, if only a small percentage of individuals have played the role of "Concept Creator", the organization can then determine (a) whether this is an appropriate role for its context and (b) if it is, can the organization identify more opportunities for systems engineers to play this role.

Organizations may also choose to use the role descriptions and profiles to increase clarity across the workforce in several ways. For example, if the roles profiles are created across the careers of individuals in key systems engineering positions (e.g., chief systems engineer, systems engineering lead, systems architect, etc.), then those patterns may provide guidance for career planning at the individual and workforce levels. This is not to say that there will only be one "right way" to grow, but clear patterns from the organization paired with self-assessment may help an individual identify which roles she would like to focus on going forward. A second way organizations may use roles is in clarifying open positions they have within the organization. In the Helix sample, it was quite rare for a position to consist of only a single role. (See Section 4.5.1 of the Technical Report, SERC-2016-TR-118.) But several participating organizations stated that current job positions – whether posted internally or externally – do not clearly outline what is expected. If the roles were clearly defined and used consistently, they could become a new model for describing what is expected in positions. That clarity would benefit both individuals seeking the roles and individuals who are selecting people to fill those roles.

5 PROFICIENCY OF SYSTEMS ENGINEERS

The proficiency model of *Atlas*, captures the knowledge, skills, abilities, behaviors, patterns of thinking, and abilities that are critical to the effectiveness of systems engineers.

> ***proficiency*** – the quality or state of knowledge, skills, abilities, behaviors, and cognition.
>
> ***proficiency area*** - grouping of related knowledge, skills, abilities, behaviors, and/or cognition.
>
> ***proficiency level*** –extent to which an individual has attained the knowledge, has the skill and ability to perform a task, or has demonstrated relevant behaviors, or cognitions.

- **Proficiency** is the quality or state of knowledge, skills, abilities, behaviors, and cognition.
- Proficiency **Areas** are groupings of related knowledge, skills, abilities, behaviors, and/or cognition.
  - Each Proficiency Area is comprised of **Categories**, which are specific types of knowledge, skills, abilities, behaviors, and cognition with shared characteristics.
    - *Some* categories are further refined into **Topics**, which are the most discrete areas of proficiency included in *Atlas*.
- For each proficiency area, there are **Levels**, which describe the extent to which an individual has attained certain knowledge, has the ability to perform a certain skill, or has demonstrated relevant abilities, behaviors, or cognition. Loosely, a scale of 0 to 10 is used to indicate the level of proficiency at the area level, where 10 indicates the highest possible proficiency.

The *Atlas* proficiency model, along with identified proficiency levels, enables a proficiency profile to be created for an individual at any point in time, as illustrated in Figures 6, 7, and 8, below. Currently, proficiency levels are documented only for proficiency **Areas**. The self-assessment proficiency tools can be found in Appendix A.

5.1 *ATLAS* PROFICIENCY MODEL

The *Atlas* proficiency model consists of six proficiency areas based on the Helix interview data, as shown in Figure 6 below.

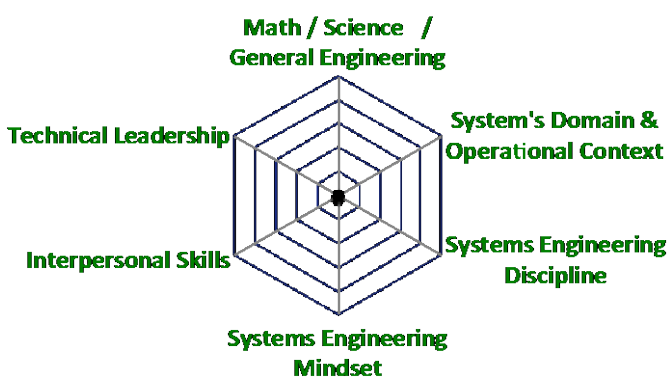


Figure 6. Proficiency Areas for Systems Engineers

1. ***Math/Science/General Engineering:*** Foundational concepts from mathematics, physical sciences, and general engineering;
2. ***System's Domain & Operational Context:*** Relevant domains, disciplines, and technologies for a given system and its operation;
3. ***Systems Engineering Discipline***: Foundation of systems science and systems engineering knowledge;
4. ***Systems Engineering Mindset***: Skills, behaviors, and cognition associated with being a systems engineer;
5. ***Interpersonal Skills***: Skills and behaviors associated with the ability to work effectively in a team environment and to coordinate across the problem domain and solution domain; and
6. ***Technical Leadership***: Skills and behaviors associated with the ability to guide a diverse team of experts toward a specific technical goal.

Proficiency areas 1 to 3 consist of primarily 'hard' or technically based skills, while proficiency areas 4 to 6 consist primarily of the 'soft' or interdisciplinary skills. The six proficiency areas in *Atlas* are further divided into categories and, in some cases, into topics, as shown in Table 4. Each of the proficiency areas is elaborated in the subsequent sections.

Table 4. Atlas Proficiency Areas, Categories, and Topics

Area	Category	Topic
1. Math / Science / General Engineering	1.1. Natural Science Foundations	
	1.2. Engineering Fundamentals	
	1.3. Probability and Statistics	
	1.4. Calculus and Analytical Geometry	
	1.5. Computing Fundamentals	
2. Systems' Domain & Operational Context	2.1. Principal and Relevant Systems	< List of Principal and Relevant Systems >
	2.2. Familiarity with Principal System's Concept of Operations (ConOps)	
	2.3. Relevant Domains	< List of relevant Domains >
	2.4. Relevant Technologies	< List of relevant Technologies >
	2.5. Relevant Disciplines and Specialties	< List of relevant Disciplines and Specialties >
	2.6. System Characteristics	< List of applicable System Types, Scales, and Levels >
3. Systems Engineering Discipline	3.1. Lifecycle	3.1.1 Lifecycle Models 3.1.2 Concept Definition 3.1.3 System Definition 3.1.4 System Realization 3.1.5 System Deployment and Use 3.1.6 Product and Service Life Management
	3.2. Systems Engineering Management	3.2.1 Planning 3.2.2 Risk Management 3.2.3 Configuration Management 3.2.4 Assessment and Control

Area	Category	Topic
		3.2.5 Quality Management
	3.3. SE Methods, Processes, and Tools	3.3.1 Balance and Optimization 3.3.2 Modeling and Simulation 3.3.3 Development Process 3.3.4 Systems Engineering Tools
	3.4. Systems Engineering Trends	3.4.1 Complexity 3.4.2 Model Oriented Systems Engineering 3.4.3 Systems Engineering Analytics 3.4.4 Agile Systems Engineering
4. Systems Engineering Mindset	4.1. Big-Picture Thinking	
	4.2. Paradoxical Mindset	4.2.1 Big-Picture Thinking and Attention to Detail 4.2.2 Strategic and Tactical 4.2.3 Analytic and Synthetic 4.2.4 Courageous and Humble 4.2.5 Methodical and Creative
	4.3. Flexible Comfort Zone	
	4.4. Abstraction	
	4.5. Foresight and Vision	
5. Interpersonal Skills	5.1. Communication	5.1.1 Audience 5.1.2 Content 5.1.3 Mode
	5.2. Listening and Comprehension	
	5.3. Working in a Team	
	5.4. Influence, Persuasion and Negotiation	
	5.5. Building a Social Network	
6. Technical Leadership	6.1. Building and Orchestrating a Diverse Team	
	6.2. Balanced Decision Making & Rational Risk Taking	
	6.3. Guiding Stakeholders with Diverse/Conflicting Needs	
	6.4. Conflict Resolution & Barrier Breaking	
	6.5. Business and Project Management Skills	
	6.6. Establishing Technical Strategies	
	6.7. Enabling Broad Portfolio-Level Outcomes	

5.2 Area 1: Math/Science/General Engineering

A good understanding of math, science, and general engineering is a critical foundation for effective systems engineers; but this understanding is largely 'assumed' in a systems engineer when joining the workforce. However, it is on this foundation that further understanding of the categories under Proficiency Area 2: *Systems' Domain & Operational Context* is built.

The *Graduate Reference Curriculum for Systems Engineering (GRCSE®)* defines the types of prerequisite knowledge individuals should have before entering a master's program in systems engineering (Pyster et al. 2015). Since limited insight was obtained from Helix data collection and analysis for this proficiency area, GRCSE is used to identify and define the categories in this area:

1.1. Natural Science Foundations: Basic concepts and principles of one of the natural science disciplines (e.g., physics, biology, chemistry, etc.); includes laboratory work that involves experimental techniques, the application of the scientific method, and comprehension of appropriate methods for data quality assurance and analysis.

1.2. Engineering Fundamentals: The nature of engineering, branches of engineering, the design process, analysis and modeling, the role of empirical and statistical techniques, problem solving strategies, and the value of standards; some level of practical experience is expected, whether through capstones, internships, or course projects. Practical experience should include the application of engineering fundamentals in a specific domain context.

1.3. Probability and Statistics: Basic probability theory, random variables and probability distributions, estimation theory, hypothesis testing, regression analysis, and analysis of variance.

1.4. Calculus and Analytical Geometry: Theory and application of differential and integral calculus methods and operations; study of techniques for describing, representing, and analyzing geometric objects (coordinate systems, algebraic models, graphing).

1.5. Computing Fundamentals: Overview of computer organization (computer architecture, operating systems, and programming languages), algorithms, and data structures; software engineering fundamentals (lifecycle models, quality, cost, and schedule issues); and development of a software unit (design, coding, and testing).

Proficiencies in Area 1: Math/Science/General Engineering may be considered as the general foundation that is provided in any undergraduate engineering degree. Advanced levels of these topics are included in the topics of Area 2, in the context of the system of concern. For an individual without a formal undergraduate degree in engineering, obtaining the proficiencies in Area 1 could happen through experience, training, or mentoring.

5.3 Area 2: System's Domain & Operational Context

The second proficiency area is *System's Domain & Operational Context*, which contains the relevant domains, technologies, disciplines, specialties, and characteristics for a given system, and the operation of that system. This proficiency area strongly corresponds to the organization and the systems that its systems engineers work on. If an individual transitions to a new system, the proficiency level may change depending on familiarity with the new relevant domains, technologies, and disciplines. The categories for this proficiency area are defined below:

2.1. Principal and Relevant Systems: *Principal* systems are those systems that are of primary interest to the organization. High levels of proficiency in those specific systems are desired by the organization. If a combat ship were the principal system, relevant systems could be submarines and aircraft carriers, which are types of combat ships.

2.2. Familiarity with Principal System's Concept of Operations (ConOps): A system's concept of operations (ConOps) of how systems in the domain are used and deliver value, especially those systems on which the individual personally works. Familiarity with the principal system's ConOps is of particular interest, though familiarity with the ConOps of other related systems may also be helpful.

2.3. Relevant Domains: *Domain* refers to the overarching area of application of the system; this includes things such as space, aerospace, marine, communication, finance, etc. Proficiency in related domains outside the primary one may enable an individual to be more effective in the primary domain. For example, experience in space systems may enable a systems engineer to work in aerospace systems more readily than an engineer who is proficient primarily in finance systems.

2.4. Relevant Technologies: Within the context of a system, there are specific technologies that are relevant. For example, on a marine system, these may be technologies such as gas turbine, radar, and sonar systems; and each technology has its own terminology, challenges, etc.

2.5. Relevant Disciplines and Specialties: Disciplines are fundamental areas of education or expertise that are foundational to a system. For example, for a communications system, electrical engineering will be an important discipline to understand, while civil engineering will be less relevant. Specialties are disciplines that support systems engineering by applying cross-cutting knowledge. Specialties include Reliability, Availability, and Maintainability (RAM), Human Systems Integration, Safety Engineering, Affordability and other related topics.

2.6. System Characteristics: Three characteristics are considered in *Atlas*:

- **System Type**: Types of systems include technical systems, social systems, human systems, physical systems, cyber systems, and any combination of these. Another classification of system types includes product systems, service systems, and enterprise systems.
- **System Scale:** Systems can be anywhere from a nano level to a distributed global or enterprise level. A generic systems engineering development process may be applicable to systems at any scale.
- **System Scope:** What can be seen as a system from one perspective, could be a subsystem from another perspective. The levels of a system could range from component/element, subsystem, system, and platform or system of systems.

5.4 AREA 3: SYSTEMS ENGINEERING DISCIPLINE

The third proficiency area is *Systems Engineering Discipline*. The categories below were developed based on data from Helix interviews about critical systems engineering knowledge and skills. The category names are taken from the *Guide to the Systems Engineering Body of Knowledge (SEBoK)* (BKCASE Editorial Board 2015). Some of the categories are further expanded into topics.

3.1. Lifecycle: *The organized collection of activities, relationships and contracts that apply to a system-of-interest during its life* (Pyster 2009). This is a roll up of knowledge about lifecycles and proficiency in specific aspects of the lifecycle. Topics 3.1.2 – 3.1.6 below, represent generic lifecycle phases in system development:

3.1.1. Lifecycle Models: *A framework of processes and activities concerned with the lifecycle that may be organized into stages, which also acts as a common reference for communication and understanding* (ISO/IEC/IEEE 15288). Lifecycle Models include the Vee model; iterative models such as the spiral development model; formal acquisition models (e.g., as defined in DoD 5000.2 2013); or less formal acquisition models (e.g., quick reaction capability or internal research and development (IR&D) models).

3.1.2. Concept Definition: *A set of core technical activities of systems engineering in which the problem space and the needs of the stakeholders are closely examined* (BKCASE Editorial Board 2016). This consists of analysis of the problem space, business or mission analysis, and the definition of stakeholder needs for required services.

3.1.3. System Definition: *A set of core technical activities of systems engineering, including the activities that are completed primarily in the front-end portion of the system design.* (BKCASE Editorial Board 2016) This consists of the definition of system requirements, the design of one or more logical and physical architectures, and analysis and selection between possible solution options.

3.1.4. System Realization: *The activities required to build a system, integrate disparate system elements, and ensure that a system both meets the needs of stakeholders and aligns with the requirements identified in the system definition stage* (BKCASE Editorial Board 2016). This includes implementation as well as integration, verification, and validation (IV&V).

3.1.5. System Deployment and Use: *A set of core technical activities of systems engineering to ensure that the developed system is operationally acceptable and that the responsibility for the effective, efficient, and safe operations of the system is transferred to the owner* (BKCASE Editorial Board 2016). Considerations for deployment and use must be included throughout the system lifecycle. Activities within this phase include deployment, operation, maintenance, and logistics.

3.1.6. Product and Service Life Management: Deals with the overall lifecycle planning and support of a system (BKCASE Editorial Board 2016). The life of a product or service often spans a considerably longer period of time than what is required to design and develop the system. This stage includes service life extension, updates, upgrades, and modernization, and disposal and retirement.

3.2. Systems Engineering Management: *Managing the resources and assets allocated to perform systems engineering, often in the context of a project or a service, but sometimes in the context of a less well-defined activity. Systems engineering management is distinguished from general project management by its focus on the technical or engineering aspects of a project* (BKCASE Editorial Board 2016). The topics contained in the *Systems Engineering Management* category are defined below:

3.2.1. Planning: Planning involves developing and integrating technical plans to achieve the technical project objectives within the resource constraints and risk thresholds. This involves the success-critical stakeholders to ensure that necessary tasks are defined with the right timing in the lifecycle in order to manage acceptable risks levels, meet schedules, and avoid costly omissions (BKCASE Editorial Board 2016).

3.2.2. Risk Management: Organized, analytic process to identify what might cause harm or loss (identify risks); to assess and quantify the identified risks; and to develop and, if needed, implement an appropriate approach to prevent or handle causes of risk that could result in significant harm or loss (ISO/IEC/IEEE 24765:2010 – SEVocab).

3.2.3. Configuration Management: A discipline applying technical and administrative direction and surveillance to: identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements (ISO/IEC/IEEE 24765:2010 – SEVocab).

3.2.4. Assessment and Control: This process involves determining and initiating the appropriate handling strategies and actions for findings and/or discrepancies that are uncovered in the enterprise, infrastructure, or lifecycle activities associated with the project (BKCASE Editorial Board 2016).

3.2.5. Quality Management: Whether a systems engineer delivers a product, a service, or an enterprise, the deliverable should meet the needs of the customer and be fit for use. Such a deliverable is said to be of high quality. The process to assure high quality is called quality management (BKCASE Editorial Board 2016).

3.3. SE Methods, Processes, and Tools: *A systems engineering method is set of activities, methods, practices, and transformations that people use to develop and maintain systems and associated products* (SEI 2007). Processes generally refer to the specific guidelines an organization develops for implementing systems engineering methods; tools refer to software programs that are designed to support systems engineering activities. The topics contained in the *SE Methods, Processes, and Tools* category are outlined below:

3.3.1. Balance and Optimization: Specialty engineers often focus on the details and optimization of their specific components of the system, but that optimization of individual components often leads to a less-than-optimal system solution. Systems engineers, therefore, have to be able to balance the desire for component optimization with the optimization for the system overall, which often requires sub-optimization for one or more components.

3.3.2. Modeling and Simulation: *A model is a simplified representation of a system at some particular point in time or space intended to promote understanding of the real system. A simulation is the manipulation of a model in such a way that it operates on time or space to compress it, thus enabling one to perceive the interactions that would not otherwise be apparent because of their separation in time or space* (Bellinger 2004). This topic represents and individual's ability to understand and perform modeling and simulation; this understanding is more fundamental than the ability to use software tools that support modeling and simulation.

3.3.3. Development Processes: Each organization has its own processes that govern the development of systems. It is important for systems engineers to understand generic systems engineering processes, but also the specific processes being used for development within the organization or domain.

3.3.4. Systems Engineering Tools: Systems engineers need to be able to utilize tools to support overall system development and to perform the systems engineering development process. Tools may include requirements management and other tools that assist with project life management (PLM).

3.4. Systems Engineering Trends: *Current and future trends in performing Systems Engineering, that modify the way systems are developed.*

3.4.1. Complexity: Complexity of a system is generally understood to exist not in a higher order scale or level of a system, but rather in the higher order of interactions between system elements, disciplines, or technologies, and the properties that emerge out of these interactions that are not present in the individual elements. One categorization of complexity includes structural complexity, dynamic complexity, and socio-political complexity; while another identifies two kinds of complexity: disorganized complexity and organized complexity (SEBoK authors, “Complexity”, 2016).

3.4.2. **Model Oriented Systems Engineering:** Model Based Systems Engineering (MBSE) is a theme that is being increasingly adopted in systems engineering, where models are used to describe various elements of systems and the systems development process. Model Oriented Systems Engineering (MOSE) goes beyond MBSE, and presents a holistic model-based approach that integrates operational, technical, programmatic and business dimensions as well.

3.4.3. **Systems Engineering Analytics:** The increasing ability to collect, store, analyze, and gain insights from large quantities of data has significantly improved the area of analytics in general. This perspective can also be applied to systems engineering, where complex phenomena within systems and systems development can be measured and analyzed.

3.4.4. **Agile Systems Engineering:** The shrinking of systems engineering development lifecycles, increasingly uncertain and rapidly changing requirements and operational environments of modern systems, has led to the development and adoption of agile systems engineering approaches.

5.5 AREA 4: SYSTEMS ENGINEERING MINDSET

The fourth proficiency area is *Systems Engineering Mindset*, which is primarily focused on patterns of thinking, perceiving, and approaching a task that are particularly relevant to systems engineers. The categories included in this area are:

4.1. **Big-Picture Thinking**: Also referred to as ‘systems thinking’ and ‘holistic thinking’, this includes the ability to step back and take a broader view of the problem at hand; this is an important and essential characteristic of systems engineers. ‘Big-picture’ could refer to a broader perspective along many different dimensions: the system as a whole including interfaces and integration, and not limited to any sub-system or component; the system while in operation, and its interactions with other systems and the operating environment; the entire lifecycle of the system, and not limited to the current stage of the system; the development program in the context of the organization and all its other development programs; the end goal or solution to the problem at hand; the perspectives of different stakeholders; and the technical as well as business perspectives. A systems engineer is usually *the* person to bring this broader perspective, while classic engineers and subject matter experts often tend to be narrowly focused on their area of interest. Systems engineers are not only called to provide this big-picture perspective themselves, but to also enable others to see this bigger picture.

4.2. **Paradoxical Mindset**: *The ability to hold and balance seemingly opposed views, and being able to move from one perspective to another appropriately*. Typically, an engineer may hold one view or the other, but rarely *both*. By having this paradoxical mindset, a systems engineer contributes value that is not usually expected from others. The opposing-concept pairs are:

4.2.1. **Big-Picture Thinking *and* Attention to Detail:** Big-picture thinking provides the broader higher-level perspective; at the same time, a systems engineer is also required to pay attention to the details of how things work and how they come together in a system.

4.2.2. **Strategic *and* Tactical:** Systems engineers need to be strategic, focused on the end result of 'vision' for the system, but also need to handle the tactical day-to-day activities and decisions required to reach that vision. They must also be able to appreciate "how what is done today is going to affect things downstream". A related concept pair is the ability to envision long-term issues but at the same time, have the desire for closure with the current situation in order to move on.

4.2.3. **Analytic *and* Synthetic:** A big-picture perspective may be associated with the ability to be synthetic, and to be able to bring together and integrate different pieces of a puzzle. However, a systems engineer also needs to be analytic and to be able to break down the big picture into smaller pieces on which others can focus and work. To do this effectively, a systems engineer needs to be able to operate at multiple levels (e.g., component, sub-system, system, system-of-systems) and multiple dimensions (e.g., various technical disciplines and stakeholder perspectives).

4.3. **Flexible Comfort Zone**: *The overall ability to deal with ambiguity and uncertainty, this involves the abilities to be open-minded, understand multiple disciplines, deal with challenges, and the ability to take rational risks*. By definition, experts possess proficiency in a specific area, which is their 'comfort zone'; and they typically do not prefer going outside that circle or comfort zone. Such experts provide value to the organization by contributing their expertise in those focused areas. However, systems engineers tend to show an ability to broaden their comfort zones, and go beyond their current boundaries and they are also comfortable doing this.

4.4. **Abstraction**: *The ability to filter out and understand the critical bits of information at the right level and to make relevant inferences*. And even with that filtered information, systems engineers need to know when to use or not use pieces of information. Such abstraction also enables systems engineers to connect and extract meaning from different streams of information; for example, to tie together information that subject matter experts of two different disciplines are providing.

4.5. **Foresight and Vision**: *The ability to foresee the remaining lifecycle of the system, the impact of current decisions, and to mentally simulate possible scenarios*. Every decision or change is likely to have an impact beyond the current confines of time or space. Particularly in early stages of a system lifecycle, and in the development of a new or unfamiliar system, foresight is a key value that systems engineers provide.

5.6 AREA 5: INTERPERSONAL SKILLS

The fifth proficiency area is *Interpersonal Skills*. Almost by definition, systems engineers do not just work by themselves at their desks all day – they interact with others. Irrespective of any formal leadership roles they may or may not play, a systems engineer is expected to be proficient in a number of interpersonal skills. While specialty engineers may be responsible for developing specific aspects of the system, systems engineers are responsible for coordinating across all of these engineers. Hence, interpersonal skills are more critical to systems engineers than they are to specialty engineers. The specific categories contained within this proficiency area are listed below:

5.1. **Communication:** Communication is critical for systems engineers since they interact with a variety of people, and is a broad category covering a wide variety of related skills and abilities. Often they are an important link between individuals and groups, both internal and external to the organization – most importantly, the customers and end-users of the system being developed. Systems engineers need the ability to clearly express their thoughts and perspectives to establish a shared common understanding.

5.1.1. **Audience:** Systems engineers need to communicate with a variety of direct and indirect audiences: customers; subject matter experts; program managers; vice presidents; directors; specialty engineers; problem owners; technical teams; contractors; decision makers; system testers; and others working on or with the project.

5.1.2. **Content:** The variety of content that systems engineers need to communicate can be broadly divided into three types, based on the audience they are communicating with:

1. **Technical**: Communications with disciplinary and specialty engineers and subject matter experts involve high technical content. But communications of technical issues to managers, end-users, and others who may not be interested in or who may be confused by all the technical detail, involves adequate abstraction of the technical content.

2. **Managerial**: Systems engineers often provide project status to managers and supervisors and cost-schedule constraints and expectations to technical personnel.

3. **Social**: Systems engineers need to maintain an amicable environment within a team and to interact with others in a courteous manner. Such interactions involve communications that are neither technical nor managerial in nature.

5.1.3. **Mode:** Communicating the intended content to the target audience is done through a number of different modes:

1. **Oral:** This takes various forms, depending on the audience and context. It could be one-on-one, or as part of a team, in person, or remotely.

2. **Presentation**: A special form of communications is the ability to stand in front of an audience and to deliver a presentation using appropriate aids. Further, during presentations, systems engineers tend to represent others who may not be in the room: they present customer needs and requirements to others in the absence of customers, and they present design decisions and system related issues to customers in the absence of designers.

3. **Writing and Documentation:** Written communication skills are equally critical for systems engineers; the scale, audience, and objective of the written artifact also matter. It could range from a short email to communicate status, to a detailed test plan, to internal documentation supporting a project decision, to design documents being submitted for review.

5.2. **Listening and Comprehension:** The ability to listen to others' points of views and perspectives, and to comprehend and internalize the message accurately. For systems engineers, listening begins with the customer to understand their real needs and ensure that these needs get translated into requirements. In a team environment, systems engineers need to listen to the views and perspectives being offered: from designers, subject matter experts, and others.

5.3. **Working in a Team:** Systems engineers tend to be part of many teams during the lifecycle of the system; further, systems engineering by itself is typically not performed by an individual, but rather by a team. Hence, team dynamics and synergy are key to the functioning of a systems engineer.

5.4. **Influence, Persuasion, and Negotiation**: It is critical for every systems engineer, not just those in formal leadership positions, to have the skills needed to make a point and to successfully obtain buy-in. In many situations, systems engineers contribute a perspective that is different from that of others: a focus on the overall system and on customer's needs. In such situations, it requires influence, persuasion, and negotiating skills for systems engineers to enable others to see the bigger picture on which they need to focus.

5.5. **Building a Social Network:** A systems engineer needs to be a 'people person', and build a social network of professional acquaintances. Such a network becomes a valuable resource for systems engineers to tap into, because they are not expected to know answers to all problems, but rather be able to find someone who has the expertise and ability to solve the problem.

5.7 AREA 6: TECHNICAL LEADERSHIP

The sixth and final *Atlas* proficiency area is *Technical Leadership*. It is common and natural for systems engineers to play leadership roles at many levels within an organization. The specific categories contained within *Technical Leadership* are listed below:

6.1. **Building and Orchestrating a Diverse Team**: *The ability to identify, build, and effectively guide or coach a team comprising individuals with diverse expertise, perspectives, and personalities*. While organizational titles may vary, it is most often a systems engineer who is the leader of the team that is charged with delivering the system. The systems engineer needs to fully know each of the team members: their strengths, weaknesses, capacities, capabilities, limitations, personalities, expertise, and working styles. The systems engineer plays the roles of coach, guide, and teacher to develop the team's capabilities and to orchestrate it to perform the required tasks. Individual leadership styles could vary, but the overall objective of is to empower the team, to instill confidence, and to help them to deliver the solution and to be successful. Another key aspect of handling a team is the ability to delegate – the leader needs to build enough trust in the team to be able to delegate with confidence.

6.2. **Balanced Decision Making and Rational Risk Taking:** Solving a problem requires a systems engineer to take a number of balanced decisions considering a variety of factors, constraints, perspectives, and objectives; as well as the implications of these decisions and their scope of impact. An additional challenge is that most often, all the required information may not be readily available. The ability to make such decisions also requires the systems engineer to be comfortable in dealing with ambiguity and uncertainty and to be able to take rational, calculated risks.

6.3. **Guiding Stakeholders with Diverse/Conflicting Needs:** This includes the ability to manage all the internal and external stakeholders, and to keep the team focused on their needs, especially those of the end user or customer. The systems engineer is uniquely positioned to interact with many stakeholders of the system – both external and internal to the organization. Being this "touch point" person, the systems engineer needs to deal with multiple personalities, behaviors, organizations, and cultures.

6.4. **Conflict Resolution and Barrier Breaking**: Conflicts are bound to rise in a variety of scenarios – within the team; within the organization – between the technical side and business side of the organization; as well as with outside the organization. As a leader, the systems engineer must resolve these conflicts while keeping the system goals in mind. In some cases, conflicts arise due to the existence of barriers, which may be related to the organizational culture, processes, team personalities, or other situations that could prevent an individual or team from getting their work done. The systems engineer needs the ability to break these barriers.

6.5. **Business and Project Management**: Depending on the way roles and titles are defined within an organization, a systems engineer's responsibilities may overlap with what may be seen as 'project management' responsibilities. Even if there is no overlap, a systems engineer is expected to handle a variety of business and project management activities including accounting, budget, cost estimation, schedule, work breakdown, and profit. The systems engineer must also be cognizant of the business impact of technical decisions that are taken.

6.6. **Establishing Technical Strategies:** Systems engineers must fearlessly and creatively guide the establishment of new capabilities and transformations (e.g., to migrate to Cloud Infrastructure, or to establish a new information service architecture, or to enable transition to a DEVOPS model). Senior systems engineers need to be able to support the organization in the development of overarching technical directions and support the development of technical roadmaps that establish a vision to support the strategy.

6.7. **Enabling Broad Portfolio-Level Outcomes:** Along with the development of strategies to guide strategic technical investments, systems engineers should provide the broad perspective necessary to enable technical success not only on individual projects but across projects and programs to enable advancement across the technical portfolio.

5.8 IMPLICATIONS FOR USE – TAILORING

As demonstrated in Table 4, there is a clear expectation that some tailoring will occur for proficiency assessment to maximize its utility. This is true for both individuals and organizations. Individuals may tailor the model specifically to what they have done – but should be mindful that all of the areas they have not touched are possible areas for future exploration. Organizations, likewise, could tailor the model before distributing it to the workforce, so that only areas that are deemed critical to the organization are captured. For example, some of the natural science foundations may not be common in a given domain and some disciplines or technologies will be considered more relevant than others. It is important to remember that tailoring may not be specific to just an organization, but also to specific programs or systems. For example, an organization that engineers financial IT systems as well as critical infrastructure systems may have different expectations and needs for those different domains.

Table 5 provides two examples of how the proficiency model could be tailored for an organization, based on the primary systems domain for each organization. Note that where <no tailoring> is listed, this indicates that the Helix team expects that either an organization will be able to use the proficiency model exactly as defined, with no tailoring required, or that for purposes of this example, no specific tailoring has been identified.

Table 5. Tailoring the Atlas Proficiency Model

Area	Category	Company 1: Defense Aerospace	Company 2: Medical Devices
1. Math / Science / General Engineering	**1.1.** Natural Science Foundations	Physics considered most critical	Chemistry and Biology considered most critical Physiology added as a Foundation
	1.2. Engineering Fundamentals	<no tailoring>	<no tailoring>
	1.3. Probability and Statistics	<no tailoring>	<no tailoring>
	1.4. Calculus and Analytical Geometry	Both are considered critical	Considered less critical than Probability & Statistics
	1.5. Computing Fundamentals	Considered less critical than the other categories	Considered critical for integration with Electronic Health Records (EHRs)
2. Systems' Domain & Operational Context	**2.1.** Principal and Relevant Systems	Air-breathing jet engines Military aircraft	Magnetic Resonance Imaging (MRI) X-Ray Computerized Tomography (CT)
	2.2. Familiarity with Principal System's Concept of Operations (ConOps)	Expectations about the level of familiarity may differ (e.g. understanding basic in-flight operations)	Expectations about the level of familiarity may differ (e.g. actual experience in a clinical setting to understand use cases, how system fits within the healthcare environment, where its use may fit in an overall process, etc.)
	2.3. Relevant Domains	Aerospace	Healthcare
	2.4. Relevant Technologies	Radar Sonar Navigation Systems	MRI X-Ray CT
	2.5. Relevant Disciplines and Specialties	Mechanical Engineering Electrical Engineering Aerospace Engineering Software Engineering Thermodynamics Aerodynamics Ergonomics	Electrical Engineering Mechanical Engineering Biomedical Engineering Software Engineering Ergonomics Radiation Safety
	2.6. System Characteristics	System level design with understanding of the system of systems in the operational environment	Systems of systems level design enabling integration with other medical devices and healthcare IT systems
3. Systems Engineering Discipline	**3.1.** Lifecycle	• V-lifecycle approach emphasized • Organization not involved in in-service operation and maintenance (full handoff after delivery)	• Spiral/Incremental Development lifecycle model emphasized • Organization heavily involved in in-service operation and maintenance
	3.2. Systems Engineering Management	<no tailoring>	<no tailoring>
	3.3. SE Methods, Processes, and Tools	• Heavy emphasis on modeling and	• Heavy emphasis in optimization for

Area	Category	Company 1: Defense Aerospace	Company 2: Medical Devices
		simulation • Emphasis on operational safety	patient safety
	3.4. Systems Engineering Trends	• Model Oriented Systems Engineering	<no tailoring>
4. Systems Engineering Mindset	4.1. Big-Picture Thinking	<no tailoring>	<no tailoring>
	4.2. Paradoxical Mindset	• Balance of Methodical and Creative heavily weighted	• Paradoxical mindset heavily weighted
	4.3. Flexible Comfort Zone	<no tailoring>	<no tailoring>
	4.4. Abstraction	<no tailoring>	<no tailoring>
	4.5. Foresight and Vision	<no tailoring>	<no tailoring>
5. Interpersonal Skills	5.1. Communication	<no tailoring>	<no tailoring>
	5.2. Listening and Comprehension	<no tailoring>	<no tailoring>
	5.3. Working in a Team	<no tailoring>	<no tailoring>
	5.4. Influence, Persuasion and Negotiation	<no tailoring>	<no tailoring>
	5.5. Building a Social Network	<no tailoring>	<no tailoring>
6. Technical Leadership	6.1. Building and Orchestrating a Diverse Team	<no tailoring>	<no tailoring>
	6.2. Balanced Decision Making & Rational Risk Taking	<no tailoring>	Risk is viewed negatively by this highly safety-conscious organization; this becomes focused on decision making.
	6.3. Guiding Stakeholders with Diverse/Conflicting Needs	<no tailoring>	<no tailoring>
	6.4. Conflict Resolution & Barrier Breaking	<no tailoring>	<no tailoring>
	6.5. Business and Project Management Skills	• Project management is treated as a distinctly separate discipline from systems engineering in this organization. There is cultural pressure not to include this as a "systems engineering" proficiency.	<no tailoring>
	6.6. Establishing Technical Strategies	• N/A (Systems engineers do not set the technical strategy for the organization)	• Only expected for senior systems engineers
	6.7. Enabling Broad Portfolio-Level Outcomes	• N/A (Systems engineers do not set the technical strategy for the organization)	• Only expected for senior systems engineers

Table 5 is only a basic example, but demonstrates that tailoring can include the identification of specific proficiencies that are of critical interest to the organization – particularly in Proficiency Areas 1 and 2, which are expected to be heavily tailored – and the emphasis or de-emphasis of categories based on the organizational context. The examples for categories 6.6 and 6.7 also demonstrate that the organization can help to set expectations about categories that are critical only at certain seniority levels.

5.8.1 Implications for Use by Individuals

The Helix team has assisted over 100 individuals in completing self-assessments based on these proficiencies and dozens of others have completed self-assessments in their organizations without the team's involvement.

During some of the Helix interviews in 2015 and 2016, interviewees were asked to self-evaluate their level of proficiency based on the *Atlas* proficiency model, at the Area level. Generally, interviewees evaluated themselves on a level of 1 to 10, where 1 was 'least proficient' and 10 was 'most proficient'. This was a subjective scale and hence when someone placed themselves at an 8 for a proficiency area, for example, it was based on their personal interpretation on what it meant. These self-evaluations – and subsequent discussions on why interviewees scored themselves in a particular way – are expected to provide insights in future research towards defining those objective scales.

Interviewees were asked to evaluate their proficiencies at two points in time: (1) at the time of the interview, and (2) at the start of their career. This enables a proficiency profile to be plotted, as illustrated in Figure 7.

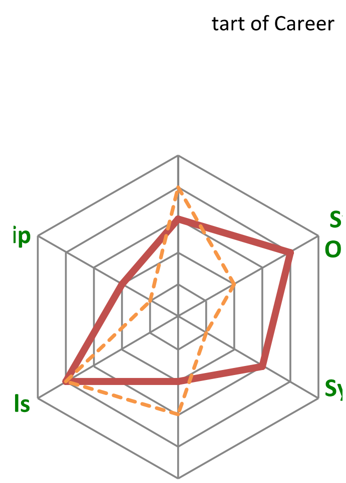


Figure 7. Proficiency Profile of an Individual

The proficiency profile is not meant to be exact since the self-evaluations are subjective, and individuals may have over-evaluated or under-evaluated themselves. Also, 'Start of Career' could be as recent as five years ago for one individual or twenty-five years ago for another. However, this exercise enables a discussion around the relative strengths in specific proficiencies; how proficiency levels changed over time; and what factors or forces caused or enabled those changes.

The primary intent of *Atlas* is not to just understand the current state of effective systems engineers, but to support the development of future systems engineers who will be effective. From a proficiency perspective, it would mean setting target levels for proficiency areas, as illustrated in Figure 8.

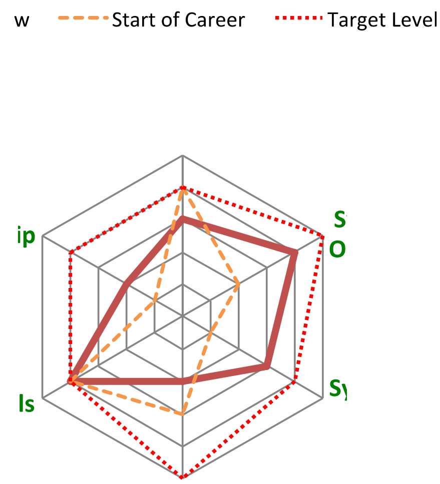


Figure 8. Proficiency Profile with Target Levels

Identifying target levels for the proficiencies will depend on the roles or positions that that individual aspires to play in future. For a junior or mid-level systems engineer, the target level could be based on the proficiency profile of a Chief Systems Engineer (CSE). This profile of a CSE is also influenced by the expectations of the organization. Having proficiency profiles, including target levels, similar to what is shown in Figure 8, would enable individuals to identify those proficiencies that need to be developed and by how much. Individual systems engineers could then plan their career development in a more focused and intentional manner, towards a specific goal.

	Math / Science / General Engineering	System Domain	Systems Engineering Discipline	Systems Engineering Mindset	Interpersonal Skills	Technical Leadership
#1	2	6	3	-1	2	2
#2	1	5	3	0	2	3
#3	-2	5	6	6	5	7
#4	-4	9	8	4	6	8
#5	2	5	5	6	3	6
#6	-2	6	6	2	2	2
#7	-2	4	6	4	2	4
#8	-1	4	6	0	1	3
#9	0	7	9	0	5	6
#10	1	2	5	2	3	3
#11	-2	4	8	0	2	2
#12	2	6	6	5	4	6
#13	1	3	6	3	3	5
#14	-1	6	7	4	2	6
#15	-1	8	7	3	4	5
#16	2	8	6	3	2	5
#17	1	5	5	3	3	5
#18	-5	5	5	4	4	5
#19	1	5	7	2	3	4
#20	-2	6	5	2	2	7

Scale
10
9
8
7
6
5
4
3
2
1
0
-1
-2
-3
-4
-5

Figure 9. Example Workforce Profile: Change in Proficiency Levels of Individuals

Figure 9 illustrates a typical distribution of changes in proficiency levels in systems engineers who participated in Helix interviews. Again, the intent is not to perform a detailed statistical analysis, but to use the information to gather insights for career development:

- The *Math/Science/General Engineering* area is one where many systems engineers said their proficiency levels dropped during their careers. One of the main reasons stated was that they were not using those skills nearly as often (or at all) in their current roles as they did at the start of their careers.
- Systems engineer #4 saw big improvements in the *System Domain*, *SE Discipline*, and *Technical Leadership* areas. Insights into the factors that contributed to those improvements will benefit others who wish to improve those proficiencies.
- Systems engineers #9 and #11 saw big improvements in the *SE Discipline* area, but did not have any change in their level of proficiency in the *SE Mindset* area. Exploring the reasons for this could reveal fresh insights.
- Overall, improvements in the *Interpersonal Skills* area are observed to be relatively modest. It would be useful in future research to explore the reasons behind this.

5.8.2 Implications for Use by Organizations

Developing the career of an individual systems engineer necessitates the concerned individual to make decisions and take required actions. However, they can be done only in the context of the organization. For example, an individual may identify the need for a master's degree in systems engineering as critical to developing some much-needed proficiencies. But if the organization does not encourage or enable its employees to pursue higher education, that systems engineer may not be able to obtain a master's degree while employed in that organization. Hence, there is a critical role that an organization plays in developing the careers of individual systems engineers.

There are insights that an organization may be able to obtain by studying the collective proficiency profiles of its systems engineers. Figure 10 shows the self-assessment of the same 20 random systems engineers included in Figure 9, for all six of the *Atlas* proficiency areas, highlighting the strongest (green) and weakest (red) proficiency areas.

	Math / Science / General Engineering	System Domain	Systems Engineering Discipline	Systems Engineering Mindset	Interpersonal Skills	Technical Leadership
#1	8	8	7	7	7	7
#2	6	6	4	8	7	7
#3	4	7	7	8	9	9
#4	5	10	8	9	7	8
#5	7	7	5	9	8	6
#6	4	8	8	9	8	8
#7	6	6	8	6	6	6
#8	7	8	8	9	8	9
#9	8	8	9	9	8	7
#10	5	8	9	9	9	8
#11	4	6	8	6	6	6
#12	6	8	8	8	8	8
#13	8	9	8	10	7	7
#14	7	8	8	8	7	6
#15	6	9	8	9	7	8
#16	6	9	7	9	8	7
#17	8	7	8	7	7	8
#18	4	8	7	9	8	8
#19	7	7	9	6	8	9
#20	6	8	9	8	9	9

Figure 10. Example Workforce Profile: Strongest and Weakest Proficiencies of Individuals

If the systems engineering population represented in Figure 10 were to belong to a single systems engineering team or group, studying it could help the team or group recognize workforce development issues as well as opportunities as identified below:

- *Math/Science/General Engineering* is the weakest area for most systems engineers. Of greater interest than identifying reasons for this trend is exploring the impacts of this on the organization's systems engineering capability.
- The *SE Mindset* and *SE Discipline* areas are the strongest for many systems engineers, but also the weakest for some. There may be an opportunity here to establish some mentoring initiatives focusing on these proficiency areas.

The *Technical Leadership* area is a mixed bag, with almost equal number of systems engineers saying it is their strongest or weakest proficiency area. Based on further exploration, a training course could be established within the organization focusing on the specific aspects of technical leadership where systems engineers feel they are the weakest.

6 FORCES THAT IMPACT THE PROFICIENCY OF SYSTEMS ENGINEERS

The three most important forces that significantly impact the proficiency of systems engineers are *Experiences*, *Mentoring*, and *Education & Training*, in that order. These forces are generated by a combination of personal and organizational initiatives. The application of these forces is the primary way by which proficiencies of an individual are developed, as illustrated in Figure 11 below.

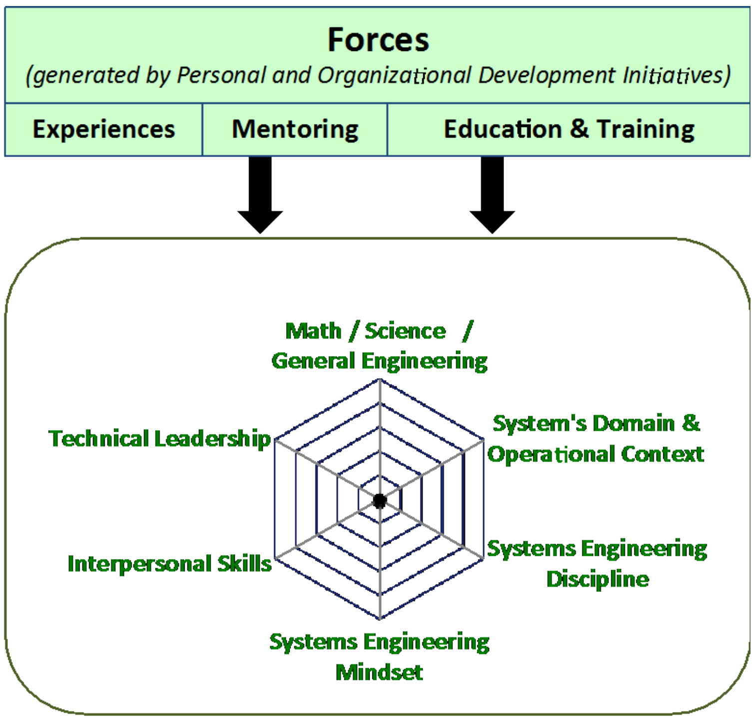


Figure 11. Forces and Proficiency

Insights into these forces that were identified from Helix data, and their relevance and importance for systems engineers, are discussed in the Technical Report (SERC-2016-TR-118) Sections 4.5.1-4.5.3.

6.1 FORCE 1: EXPERIENCES

Experiences are considered the most critical factor contributing to the development of proficiencies and to the overall growth of systems engineers. However, it is the characterization of these experiences that provides insight into how they impact proficiencies over time. Considering experiences as a force, each of these dimensions contributes to increasing one or more areas of proficiency. Experiences can also impact the personal characteristics of an individual. *Experiences*, as considered in *Atlas*, includes experiences along the following characteristics:

- **Relevance:** Every experience cannot be considered to be relevant to the development of systems engineers. A 'relevant' position is one that enables a systems engineer to develop the proficiencies critical to systems engineering. A 'systems engineering' position is one where the individual's primary focus was on SE activities.

- **Position:** Every systems engineer who is employed at an organization fills a position that is established by the organization; that organization also defines the roles and responsibilities to be performed. Helix considers position as a 'unit of measure' for experience, since most of the characteristics of experience is in the context of the position that is being held.
- **Chronological Time:** The amount of time spent in any particular position or in performing a role.
- **Number of Organizations:** The number of different organizations that an individual has worked at, not counting internal movement within an organization across departments or divisions, reflects the variety of experiences that one may possess. In large corporations that have multiple business units, or in situations where there are mergers and acquisitions, this number may not be a good indicator of the variety of experiences.
- **Organizational Sectors:** There are many differences in the general characteristics of an organization based on its sector. In *Atlas*, three organizational sectors are identified: government, industry, and academia.
- **Roles:** The 16 roles identified in *Atlas* are described in Section 3.
- **Lifecycle Phases:** Generic systems engineering lifecycle phases considered in *Atlas* are described in Section 5.4. The titles and descriptions of lifecycle phases or stages may vary across different systems engineering processes and frameworks available in literature or in use at an organization.
- **Systems:** There are many aspects to the types of systems on which a systems engineer could work. Working across these different categories provides valuable experience to an individual systems engineer.
    - **Domain:** This is the primary area of application for the systems being worked on. However, there are many domain categorizations; some domains also relate to industry sectors.
    - **Type:** Product systems, service systems, and enterprise systems are three major types of systems, depending on the nature and composition of the system of interest. System of systems is another paradigm in systems engineering, and could be a combination of one or more types of systems.
    - **Level:** A systems engineer could work on various levels of a system: component/element, subsystem, system, and platform or system of systems.

6.2 Force 2: Mentoring

Mentoring (or mentorship) is a relationship between two individuals: a mentor possesses more experience and knowledge and shares these with a mentee for the mentee's personal development. The effectiveness and derived value of the mentoring relationship is dependent on the individuals involved, but is also influenced by the organization which derives value out of a mentoring relationship as well.

6.2.1 What is Mentoring?

Mentoring means different things to different individuals and in different organizations. Common characteristics of mentoring are discussed below.

- Two individuals are involved in a mentoring arrangement: a mentor and a mentee (also referred to as a protégé).
- The mentor is usually *senior* when compared to the mentee in age, experience, and/or expertise.
- Primarily, the mentor *gives* and the mentee *receives*.
- The mentor-mentee relationship is a many-many relationship: a single mentor can have multiple mentees, and a single mentee can have multiple mentors – concurrently or spread over time.
- Mentor-mentee interactions typically happen over an extended period of time at varying frequencies.

There are also some differences and contradictions in the understanding of mentoring.

- Some use the term mentoring to describe any interaction with any co-worker in the organization that would provide any advice or guidance to handle the problem at hand.
- Some consider mentors to be synonymous with subject matter experts (SMEs) who are consulted for their expertise on an as-needed basis only. In contrast, some consider it mentoring only if the mentor is a senior person, and only if there are regular interactions between the mentor and mentee over an extended period of time.
- When the mentor and the mentee are of the same seniority in terms of age, years of experience, or level of expertise, some still consider it to be a mentoring relationship, while some others consider it to be a peer-peer relationship and not a mentoring relationship.
- Some distinguish between the concepts of coaching and mentoring: coaching is related to providing advice and guidance on solving a specific technical problem, while mentoring on the other hand, has neither a set beginning or end to the relationship, nor is related to a specific event.

6.2.2 Mentoring Arrangements

Mentoring arrangements can either be formal or informal, depending on the level of engagement of the organization in establishing and sustaining the mentoring relationship. The two types of mentoring arrangements may be summarized as below:

- **Formal**: The organization plays an active role in establishing the mentor-mentee relationship, and also lays down guidelines for maintaining that relationship. Usually, organizations require that objectives and expectations for the mentor and the mentee be stated explicitly. The relationship and its progress tend to be monitored by the organization.
- **Informal**: The participating individuals establish the mentor–mentee relationship by themselves: either a mentor adopts a mentee or a mentee seeks out a mentor, and the relationship is established. Formal objectives or expectations are usually not stated explicitly, but it is considered good practice to establish these in some form at the start of the relationship. The organization plays a less active role in informal mentoring. It is upon the mentor and the mentee

to establish and drive the relationship.

6.2.3 MENTORING FOCUS

Depending on what the mentoring is about, interviewees mentioned three types of mentoring:

- **Career Mentoring:** The mentor provides advice on career-related issues: helps identify career goals and the paths leading to that goal. The mentor could be from another group or division in the organization. Mentees are also groomed on management and leadership related topics.
- **Technical Mentoring:** The mentor typically provides advice on the technical details of the system being engineered. The mentor teaches lessons that are typically not found in textbooks and provides crucial insights on technical tools and processes. The mentor also acts as a subject matter expert, answering questions mentees might have on the subject, the system, or the program.
- **Organizational Mentoring:** While closely related to career mentoring, in organizational mentoring the mentor provides information about the organization: its culture, its procedures, and its policies. This is especially critical to a new employee.

6.2.4 BENEFITS OF MENTORING

In any typical mentoring arrangement, the mentor 'gives' and the mentee 'receives'. Therefore, such an arrangement is expected to be most beneficial to the mentee. However, there are benefits to the mentors as well. In addition, the organization also stands to benefit. Whenever an organization establishes a formal mentoring initiative, it usually expects to derive some benefit out the mentoring arrangements. However, the benefits to the mentee, to the mentor, or to the organization are conditional, and should not be taken for granted.

- **Benefits to Mentees**: The mentee gains significantly through mentoring. Most interviewees identified mentoring as a critical factor that increases the effectiveness of systems engineers. The biggest benefit to mentees of mentoring is the relationship they establish with their mentors over the span of their careers; most other benefits of mentoring are enabled through the mentor. Through their mentors, employees often get exposed to opportunities within the organization that may not be visible otherwise. During mentoring, mentees often receive important lessons from their mentors, which have made a significant impact in their careers. Finally, mentoring enables a mentee to build a strong professional network.
- **Benefits to Mentors:** Though the mentee stands to benefit the most, the mentor also benefits by mentoring, which tends to motivate the mentor to engage in a mentoring a relationship. Many considered mentoring to be an important part of their jobs; helping rising stars and teaching younger engineers what to do was motivation enough for most mentors. In organizations where mentoring is acknowledged, mentors get recognized for their efforts, for example in annual performance evaluations. Some mentors considered mentoring to be a means of reducing their workload when a mentee is able to take responsibility for a portion of the work. Finally, mentoring can be a critical way to groom a successor. This was particularly heard from senior systems engineers, but could be relevant at any stage in the career.
- **Benefits to Organization**: Effective mentoring not only benefits the mentees and mentors

involved in the relationship, but also the workforce as a whole. When this happens, the organization at large benefits as well. Good mentoring was seen as one of the most efficient ways enable effective knowledge transfer from the senior members of the workforce to more junior members. Through the feedback from mentors, organizations can also identify high-potential engineers who are being mentored. Effective mentoring can significantly reduce the time taken for new employees to get oriented to their jobs, making them effective more quickly. Effective mentoring was also seen as a mechanism for improving employee retention; when an individuals feel they have someone “in their corner” who is helping them on the job and shepherding their careers, they are more likely to feel valued and less likely to look for opportunities outside the organization.

6.3 FORCE 3: EDUCATION & TRAINING

Education plays two key roles in the development of systems engineers:

1. It provides the foundation knowledge to support engineering-related work. Typically, this takes the form of undergraduate education in an engineering discipline, technical field, or physical science.
2. Graduate level education is an avenue to develop more advanced skills, explore more in-depth knowledge, and help systems engineers grow as they move through their careers.

In addition to formal academic programs leading to undergraduate and graduate degrees, there are graduate certificates that individuals obtain, in an area that is closely related to their work. Some systems engineers go on to obtain doctoral degrees as well.

Systems engineers typically start their careers after obtaining an undergraduate degree, while graduate degrees may be obtained immediately after an undergraduate program or after a few years of professional work. Any formal degree directly improves proficiency in the relevant areas and categories. Any undergraduate degree in engineering typically provides much of the *Math/Science/General Engineering* proficiency in addition to the relevant categories under the *Systems’ Domain & Operational Context* proficiency area. Graduate degrees add to relevant proficiencies; much of the formal systems engineering education happens at the graduate level.

While academic programs are typically offered by a university, there are a number of tailored training programs that organizations offer their employees. These trainings are more focused on building specific skills that are required for them to perform their work and are typically offered short-term. The topics vary widely across organizations, with some training focused on the technical aspects of systems development, other training focused on organization-specific approaches and processes, and still other training focused on leadership or interpersonal skills. Each type of training has a role in the development of proficiency.

Among the six proficiency areas in *Atlas*, *Math/Science/General Engineering*, *System’s Domain & Operational Context*, and *Systems Engineering Discipline* may be considered to be ‘hard’ proficiencies at large, while *Systems Engineering Mindset*, *Interpersonal Skills*, and *Technical Leadership* may be considered to be ‘soft’ proficiencies at large. Formal education typically improves the hard proficiencies, but training could improve both hard and soft proficiencies.

In general, education or training results in an initial, single increase in proficiency. Additional changes over time are then the result of applying the knowledge or skills gained through this force in a real-world setting; i.e., through experiences utilizing the outputs of the education or training.

Characteristics that would be identified for relevant Education and Training would include:

- Type (education or training)
- Duration
- Date/Type of Completion (graduation date for an academic degree, course completion date for a single educational or training course)
- Subject matter covered
- Expected and/or Actual Outcomes, particularly in the context of expected changes to a systems engineer's proficiency after completion.

6.4 IMPLICATIONS FOR USE

The assessment of the Forces can provide insights in several ways. For individuals, review and clear analysis of past experiences, mentoring, education, and training can help an individual more clearly understand and document the rationale for her proficiency assessment. This analysis can also help an individual determine any critical gaps, such as areas of experience that are lacking, the need for a mentor, etc. and these insights can enable future planning. For an organization, reviewing the forces data for the existing workforce can likewise enable identification of common gaps in the workforce and these can inform decisions about organizational initiatives. For example, if there is a common workforce gap related to a specific role or phase of the systems lifecycle, the organization might develop a rotational program to enable more individuals to gain experience in these areas.

The tools for assessing the forces over time are contained in the career path approach. See Section 9 of this report and the associated tools in Appendix A for more information.

7 Personal and Organizational Characteristics

Personal characteristics and organizational characteristics can either enable or inhibit a systems engineer's ability to deliver value. They also impact the effects of the forces that influence the effectiveness of the systems engineer. However, it is also possible for the characteristics to be influenced by the forces, as illustrated in Figure 12.

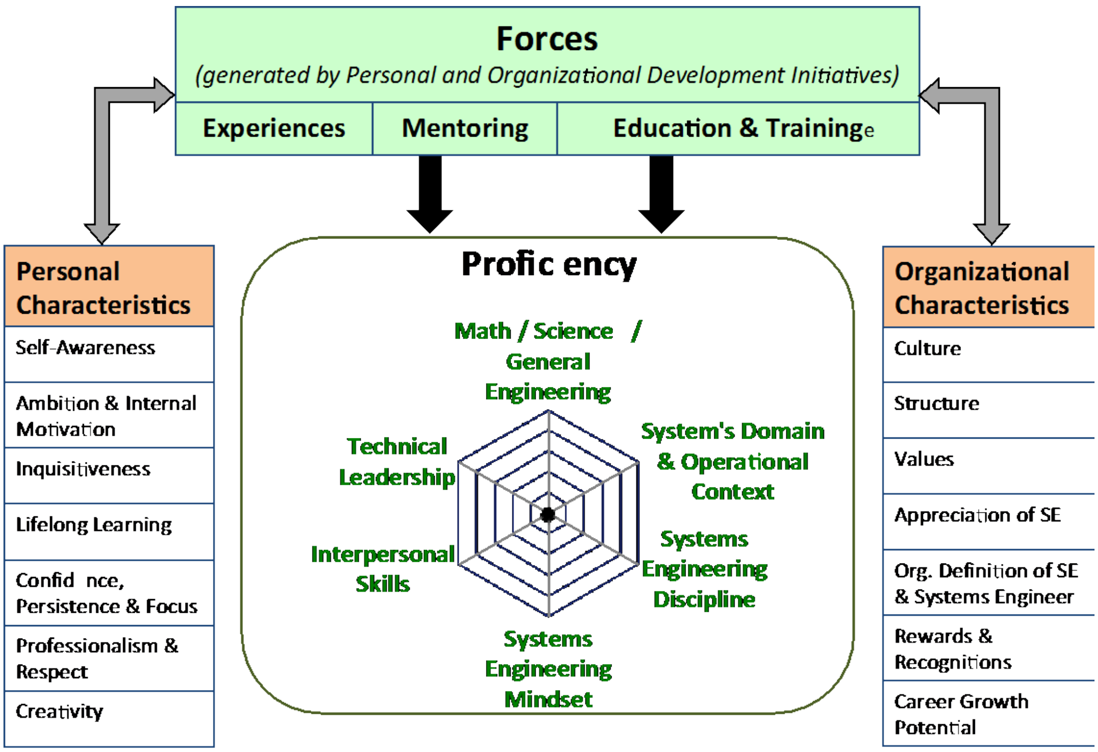


Figure 12. Forces, Proficiency and Characteristics

7.1 Personal Characteristics

Personal characteristics relate more to the personality of an individual, which implies:

- While forces that are generated through personal and organizational initiatives are expected to have a direct and significant effect on levels of proficiencies, the effect of those forces on personal characteristics is expected to be less.
- Personal characteristics are key enablers for forces to impact and grow proficiencies. Conversely, the lack of some personal characteristics may slow down or even prevent growth of some proficiencies.
- There is not enough evidence to state whether the personal characteristics are innate or learned. However, it appears that they can be influenced or improved (examples not specific to engineering include: Freshwater 2002, Koen et al. 2012, and Coldstream 2006).

Personal characteristics tend to be a differentiator between individual systems engineers. For example, two individuals with similar educational backgrounds and experiences undergoing the same training program may accrue different levels of benefits. Significant personal characteristics are:

- **Self-Awareness:** The ability to self-reflect and become aware of one's own strengths, weaknesses, knowledge, and lack thereof.
- **Ambition and Internal Motivation:** The desire to reach high career positions, and the ability to draw motivation and energy from within in order to accomplish those high ambitions.
- **Inquisitiveness**: Possessing a high level of inherent curiosity, wanting to know more and have a 'hunger for knowledge'.
- **Lifelong Learning:** Always looking to learn and to keeping abreast with latest developments in related disciplines and systems, irrespective of seniority or position.
- **Confidence, Persistence and Focus:** Possessing the confidence to interact with stakeholders irrespective of their relative seniority or positions; the ability to stand firm and not give-up; and the ability to remain focused on the success of the overall system.
- **Professionalism and Respect:** Being professional in the conduct, mannerisms, and behaviors; and treating others with respect, recognizing that other experts may possess more knowledge and experience.
- **Creativity**: Combination of left-brain – right-brain thinking bringing an artistic perspective to technical issues. Systems engineers need to be disciplined, organized, diligent, methodical, and process-oriented in their approach; they need to stay focused on the end-result and the path towards that. However, they also need to be creative in thinking through problems at hand and arriving at solutions without compromising the disciplined approach. Systems engineers should be flexible and adaptable in order to effectively respond to change and unexpected disruptions.

7.1.1 Implications for Use

There is debate not only in systems engineering but in the broader workforce development literature about whether these types of characteristics can be learned/trained or whether they are inherent. This question was asked of many systems engineers in the Helix sample, and none could give a clear perspective – though the most common answer was, "It is a bit of both." Because *Atlas* primarily reflects the data from Helix, the Helix team does not delineate *how* a systems engineer gains these characteristics, only *that* they were consistently stated as critical for effectiveness. Literature from medicine (e.g. Freshwater 2002), workforce development (e.g. Koen et al. 2012), and higher education (e.g. Coldstream 2006) indicates that it is possible to make some improvement on characteristics such as self awareness and curiosity. In future, the Helix team will examine this literature to determine if there are clear recommendations on approaches to improving these personal characteristics.

There are many different frameworks that speak to a variety of these characteristics. The Myers-Briggs Type Indicator (MBTI) test is common in industry and highlights 16 characteristics of personality (The Meyers & Briggs Foundation 2016), some of which align with the Personal Enabling Characteristics identified here. The DISC personality profile assesses behavior, temperament, and personality – again with some characteristics that overlap with *Atlas 1.0*, for example "Contentiousness" which aligns well with internal motivation. (Harrison 2016) Likewise, any individual who voluntarily explores their own personality using these types of assessments demonstrates self-awareness – or the desire for self-awareness – to some extent.

There are no assessments that exactly match the *Atlas* Personal Enabling Characteristics. The questions, then, for an individual or organization using *Atlas* is which existing framework makes sense and how should the results be used? Some organizations may already employ such a framework and for those that do, understanding the implications of results for systems engineers would be useful. For example, are there combinations of results that generally result in a good systems engineer within an organization? If so, these could be considered indicators of candidates for new systems engineers. They might also provide insight into the existing workforce; for example, an individual with a very different profile may need more assistance via training or mentoring than an individual who generally aligns with these indicators. Again, the Helix team does not make any specific recommendation on metrics to use at this time, but instead recommends that an organization begin to understand how personal enabling characteristics are currently playing in their systems engineering workforce and use any available data to determine if changes to the current hiring, selection, or workforce development approaches might be appropriate.

Individuals who have already participated in these assessments should consider their results in light of *Atlas* Personal Enabling Characteristics and determine whether there are any key differences that might explain their current situation and effectiveness within their organization. With this knowledge, systems engineers can determine what changes to make to improve their effectiveness.

7.2 Organizational Characteristics

There are several organizational characteristics that influence how difficult or easy it may be for a systems engineer to be effective. The first grouping of characteristics is not unique to systems engineering but provides the overarching context of the organization – these characteristics would likely influence the effectiveness of any individual in the organization, regardless of her discipline, but are still critically important to understanding the context in which a systems engineer operates. The other characteristics are specific to how an organization views, communicates about, and values systems engineering.

- **Culture, Structures, and Values:** While an organization's overarching culture, structure, and values have a much bigger impact than just on the systems engineering community, these factors certainly impact the ability of systems engineers to provide value to the organization.
    - A culture that values individual contributions over team contributions, for example, is a difficult environment for a systems engineer whose value is often realized through team coordination and interaction.
    - The way systems engineers are placed within the overall organization and how they are deployed to projects can affect performance.
    - Organizations that do state a value proposition for systems engineers tend to make systems engineering training more available and facilitate outreach with other disciplines.
- **Appreciation of Systems Engineering:** If an organization has no value proposition for systems engineers or if the value proposition for systems engineers is unclear, it raises uncertainties with individuals outside of the systems engineering community. These individuals do not understand what to expect from systems engineers or what return on investment to expect when they allocate a portion of their budget to systems engineering activities.

- **Organizational Definition of "Systems Engineering" and "Systems Engineer":** When an organization has an ambiguous definition of these terms – or no definition – it is an impediment to a systems engineer's effectiveness. In organizations lacking clear and unambiguous definitions of these terms, individuals outside of the systems engineering community form their own impression of what systems engineers do based on their personal experiences with an often limited sample of systems engineers. When the title "systems engineer" is applied loosely within an organization, it can cause tension, as people do not have clear expectations of what value a systems engineer should truly bring to a project.
- **Rewards and Recognition:** Organizations tend to have a very common and generic annual performance evaluation system; there are no specific outcomes or objectives related to the value that systems engineers provide. Organizations need a consistent means of evaluating or rewarding systems engineering practice.
- **Career Growth Potential:** In organizations where the career path for a systems engineer is obscure, the discipline is seen as less appealing than other areas where career growth and opportunity is more clearly defined.

These elements are related – for example if an organization does not define a systems engineer, it would be difficult for an individual to then understand how to progress in her career as a systems engineer and likewise it is lessens the likelihood that the organization will recognize value from systems engineering-specific efforts. This is illustrated in the example below, which reflects the Helix team's experiences with one organization.:

> *At one organization, project managers interviewed stated that when they got a "good" systems engineer, that person was critically important to helping them understand the technical vision and possibilities for a system. Good systems engineers also armed them with the information they needed to make trade-off decisions between technical capability and budget or schedule impacts. However, if they got a "bad" systems engineer, they were likely instead to feel encumbered with extra process – more work and restrictions – with no value added that they could define. Systems engineers in this organization stated that they were often viewed as "process wonks" because the only metrics their managers understood for systems engineers were related to formal process. They felt that if they did what they believed was good systems engineering, it was not valued. Instead the delivery of specific documents was instead used to assess their effectives. This did not align with their vision of what systems engineering should do. If the organization clearly communicated the expectations for and potential values provided by systems engineers, then managers, program managers, and systems engineers would all have a clearer understanding of effectiveness in that context. Then the organization could more clearly define and foster an appreciation for the benefit of systems engineering and reward them accordingly. This could result in an improvement of effective systems engineering, making the systems engineers feel more appreciated and rewarded for doing what they deem "the right things."*

For *Atlas 1.0*, the state of organizational characteristics around systems engineering are effectively tri-modal: in the sample, organizations either show *good* practices, had *no* practices, or there was some *muddle* in between. For example, most organizations did not have any standard definition for "systems engineering" or "systems engineer" and of the organizations that did have these, there was a disconnect between the organizational view and the understanding by the systems engineers in that organization. In an organization that did have clear definitions, for instance, it was common in interviews for systems engineers to report they were hearing the "official" definitions for the first time during their interview.

7.2.1 Implications for Use

For each of these organizational metrics, many frameworks currently exist for more rigorous assessment. For example, for Organizational Culture, Cameron and Quinn's "Organizational Culture Assessment Instrument (OCAI-1)" (2011) provides a view of culture based on a balance of focus (internal and external), flexibility, and stability. Harrison (1972) identifies four different types of organizational cultures (power, role, task, and person), which link the culture and structure of the organization. Maximini (2015) details how organizational culture affects the ability to implement agile methods and likewise provides an overview of several different organizational models. The Helix team does not recommend a specific organizational culture framework at this time; rather the team recommends that organizations interested in using *Atlas* select a framework from which to understand and document its culture. In organizations where a framework is already in use, this can easily be done.

Individual systems engineers, likewise, would benefit from understanding more crisply their organizational context and to examine how this context impacts their own effectiveness. It is unlikely that an individual would have the power to change his or her organizational context; however, awareness of some potential issues and their causes may enable them to adapt to better provide value within that context.

8 PERSONAL AND ORGANIZATIONAL DEVELOPMENT INITIATIVES

Personal development initiatives are what *individuals* do to improve their own effectiveness. Organizational initiatives are programs created by an organization with the express purpose of improving the capabilities of their systems engineering workforce. Personal initiatives do not include *participating* in organizational initiatives. For example, if an individual obtains a master's degree as a member of an organization-sponsored cohort, that would be considered an organizational initiative.

8.1 PERSONAL DEVELOPMENT INITIATIVES

When asked what personal initiatives they had for improving their own effectiveness, 100% of the systems engineers in the sample participated in organizational initiatives in some ways – most specifically in mandatory training or mentoring programs. Many fewer individuals had personal growth initiatives (7%) outside of the initiatives of their organizations. There were a few common approaches:

- *Individual Reading* – Some individuals reported that they spent personal time reading material related to their work; e.g., journal articles, conference papers, trade publications, relevant news or magazine articles, or books. Journal articles, conference papers, trade publications, and new articles tended to be around technical subjects – new technologies related to the systems the individual supported, classic engineering disciplines, relevant domains, or systems engineering itself (such as the INCOSE *Systems Engineering* journal or the IEEE *Systems* journal). When individuals read books for self-development, they were more commonly on non-technical topics such as technical leadership - particularly business – or interpersonal skills – particularly communication.
- *Attending conferences* – Several individuals stated that they attended conferences relevant to their work whenever possible – generally, a mix of domain-specific, classic engineering, systems engineering, or project management conferences. Individuals who attended conferences stated that their organizations sponsored their attendance, but that this was not a broad initiative; rather, their individual managers or programs helped them find funding to attend relevant events. A few individuals said that they used to attend conferences, but that funding was no longer available for these efforts and had not been for the last five years or more.
- *Online courses* – these are not full academic courses for credit that could be counted towards a degree. Those types of courses were considered education. However, a few individuals indicated that there were free courses available online; e.g., massive open online courses (MOOCs) or small, university-sponsored free courses on relevant topics. Popular topics included overviews of basic classic engineering disciplines such as electrical or software engineering, as well as risk- or decision-management, and specific technology areas. Individuals who took these courses said they were helpful to master an overview of an area, particularly on topics that were relevant to the systems on which an individual worked, but in which she did not have experience. Because these courses are not sponsored by the company, taking them is wholly dependent on individual motivation.
- *Certification* – *All* DoD organizations required an engineering certification (at the time of the Helix interviews, the Systems Planning, Research, Development, and Engineering (SPRDE) certification) for all of their systems engineers. However, a few individuals had also sought additional certification. No organization specifically sponsored external certification initiatives,

and the few individuals who had become certified said that they did not believe that it would help them in their organizations. They felt additional certification was important for them as individuals. The three types of certifications discussed were INCOSE Certified Systems Engineering Profession (CSEP); PMI Project Management Profession (PMP); and state-certified Professional Engineer (PE). Note that only the first certification is unique to systems engineering.

Of the individuals who stated they did not do anything outside of organizational initiatives, many junior and mid-level systems engineers said that they would like to, but that there are roadblocks. The most commonly stated are time-consuming work responsibilities and managers who do not support additional training. In one organization, individuals stated that they were expected to pursue training but were not given leave from their roles and were "dinged on their performance" for failing to get additional training. Most senior systems engineers who discussed personal initiatives stated that beyond reading or attending conferences, they believed building on their experiences was sufficient. However, almost 5% of senior systems engineers had at one point created training programs specifically to pass on their knowledge and experiences to younger systems engineers in their organizations.

8.2 Organizational Development Initiatives

Helix identifies 'initiatives' (both personal and organizational), as those that are intended to generate one or more the forces (experiences, mentoring, and education & training) in a direct manner. These forces, in turn, are expected to improve the proficiency of an individual systems engineer. This section presents various aspects of organizational development initiatives that were discussed during Helix interviews, with a particular focus on initiatives that are available for the benefit of the systems engineers in the organization.

The discussion presented in this section is aggregated from the 40% of all Helix interviews in which participants discussed organizational initiatives. In organizations with a larger number of Helix participants, a richer view of the organization emerged, sometimes with conflicting views presented by the participants. While these are highlighted in the discussion, the intent is not to provide an organization level analysis of initiatives.

8.2.1 Nature of Organizational Initiatives

Many features of organizational characteristics can be observed from Helix interviews:

- **Distinction between initiatives and policies**: It is not always straightforward to recognize and identify organizational initiatives, and to distinguish them from organizational practices and policies. Helix considers it an initiative if the organization plays an active role in promoting, enabling, and supporting it for the benefit of its employees. For example:
    - Some organizations provide tuition reimbursement to their employees seeking graduate degrees in related disciplines, subject to policies regarding eligibility, absence from work, etc. Typically, it is up to the individual employee and her immediate supervisor to take advantage of those policies.
    - Other organizations play a more active role in providing graduate education for their employees: they establish relations with specific universities; they establish cohorts for individual courses and/or degree programs; they provide facilities within their premises

for the universities to conduct courses; they make available organizational data for projects and dissertations; and also tend to reward employees who go through these programs with a promotion or salary raise.

- **Scope of organizational initiatives**: Some organizational initiatives are targeted at systems engineers' proficiencies, systems engineering proficiencies of the workforce, or within the systems engineering department/division. There are initiatives that are offered only to those systems engineers that meet certain eligibility criteria and not to the entire systems engineering population. These "high potential" programs are generally intended to help selected systems engineers mature more rapidly. There are also other initiatives intended for the benefit of all employees across the entire organization, which include any systems engineers; for example, some organizations will pay for any graduate education, regardless of subject. Each of these can be a benefit to a systems engineer, though programs scoped specifically to the systems engineering population tend to be more directly beneficial.
- **Influence of organizational initiatives on organizational characteristics:** While some organizational initiatives generate forces that in turn improve the proficiency levels of individual systems engineers, some other organizational initiatives improve organizational characteristics – either directly or indirectly. For example:
  - Some organizations have initiatives to identify and recruit SE talent from within the organization, and also to recognize and reward achievements of systems engineers and other employees. Such initiatives do not directly improve any of the forces, but rather the organizational characteristics.
  - Some organizations have mentoring initiatives to develop their junior systems engineers by pairing them up with senior systems engineers. Such initiatives are intended to directly benefit the mentee. However, such relationships between junior and senior systems engineers also tend to improve the environment and culture of the organization. (See Section 6.2.4 on the benefits of mentoring.)
- **Formal and informal initiatives**: By definition, organizational initiatives are formally established and deployed. However, there are also informal versions of those formal initiatives that could even co-exist with formal versions within the same organization. Some informal initiatives are also established by the organization. For example:
  - It is typical for mentors and mentees to form an informal mentoring relationship, without being explicitly directed by the organization. Such informal mentoring relationships tend to exist irrespective of the establishment of a formal mentoring initiative in that organization.
  - Some organizations offer a variety of training courses on topics of relevance, often in a classroom setting. In addition, there are also informal training and information sessions that the organization offers – as guest lectures or lunch-and-learn programs.
- **Portfolio of initiatives**: Organizational initiatives rarely exist in isolation; typically, a portfolio of initiatives is available to employees. Organizations establish individual initiatives to address various needs; and in some cases, a higher-level initiative leads to many lower level initiatives as well. For example, an organization may have mentoring and rotational programs. These may be linked, such that each new rotation pairs an individual with a new/additional mentor. An individual in the rotation program, then, not only gains skills from new work experiences, but also develops a larger network of trusted individuals on whom she can call for advice and

support.

As another example, an organization may have a goal to increase the percentage of the workforce with graduate degrees and creates an incentive program for graduate education, paying for tuition and giving an individual a number of paid hours each week to devote to study. If many systems engineers take advantage of this to gain formal systems engineering education and the organization identifies clear positive impacts, the organization may decide to partner with a university to develop a cohort program for systems engineering master's education.

8.2.2 Types of Organizational Initiatives

Participants in Helix interviews discussed the features, benefits, and shortcomings of many organizational initiatives that they had either directly participated in or have been aware of – both in their current organizations and in their previous organizations. The many initiatives mentioned, may be classified under the following types:

- **Recruitment initiatives**: These initiatives recognize systems engineering talent and bring individuals into the systems engineering fold. In some organizations, such initiatives bring in new employees from outside the organization – usually fresh graduates or others with limited experience. Other organizations have initiatives to recognize and recruit systems engineers from elsewhere in the organization, usually after a manager has identified the person as a "systems thinker".
- **Orientation initiatives**: Some initiatives are exclusively targeted at new employees to familiarize them with the organization, its processes, and the way it does systems engineering. In most organizations, a job rotation program is usually offered only to new / junior employees, offering them a glimpse into various parts of the organization before assigning them to one part of the organization. Some organizations recognize the value of such initiatives to senior employees, and extend those initiatives to them as well.
- **Experience enhancing initiatives:** Junior systems engineers grow into senior experienced systems engineers not just by the number of years they spend in an organization, but through performing in various systems engineering roles; different projects; various levels and types of systems; and different phases of a systems lifecycle. Organizations establish initiatives that are designed to effectively provide rich experiences to systems engineers. Typically, these take the form of rotational programs with specific paths depending on the types of skills to be developed.
- **Mentoring initiatives**: These initiatives are very prevalent in many organizations – either as a formal or an informal arrangement. While the primary beneficiaries of mentoring arrangements are the less experienced mentees, the more experienced mentors and the organization at large stands to benefit as well. From a Helix perspective, 'mentoring' is also identified as a force that directly impacts and enhances the proficiency of systems engineers. Section 6.2 provides additional discussion on mentoring and mentoring initiatives.
- **Education and training initiatives:** Every employee enters any organization with some level of formal education. Recognizing the value of formal education, many organizations offer many initiatives for their employees to obtain higher degrees from universities. There is also a need for employees to be trained in particular specialized topics, and organizations typically offer many training options of varying types and durations for the benefit of its employees. Various

aspects of training are discussed in Section 6.3.

- **Knowledge management initiatives**: A significant risk in many of the organizations that participated in the Helix interviews was the imminent loss of senior system engineers and their vast experiences. Many organizations have established initiatives to capture those experiences in various ways, and to store them in a readily accessible manner as when required.
- **Leadership development initiatives:** The most senior technical position that a systems engineer can achieve in an organization is that of a chief systems engineer or equivalent. Organizations tend to identify high-potential employees from its pool of junior and mid-level systems engineers, and offer them initiatives to enhance their leadership proficiencies in addition to technical proficiencies, thus enabling those systems engineers to develop in to future chief systems engineers and other senior systems engineering positions.
- **Rewards and recognition initiatives:** As a way to motivate, encourage, and appreciate the achievements of its systems engineers, organizations establish various rewards and recognition initiatives specifically for systems engineers in addition to its employees at large.

Overall, initiatives are focused on helping individuals develop additional proficiency using one or more of the forces identified in *Atlas*. For example, rotational programs are designed to increase the breadth of experiences. Apprentice programs – where an individual is paired with a more senior individual and shadows them – provides an opportunity for building proficiencies through both experiences and mentoring. Rewards initiatives generally help to identify and provide solid examples of effective systems engineers, highlighting the key systems engineering values for the organization.

8.2.3 Phases of Organizational Initiatives

Helix interview data indicates that organizational initiatives tend to have various phases. Appropriate recognition and management of initiatives across these different phases is critical for success.

- **Identifying the need:** The first step in any organizational initiative is to clearly articulate the need for one, or define the problem that needs to be solved. While there are many types of initiatives that an organization could potentially establish, it is imperative for an organization to understand why a particular initiative is required.
- **Establishing the initiative**: Once the need is recognized and the type of initiative is identified, the organization must then establish the initiative by setting up the required policies, guidance, personnel to run / manage the initiative, criteria for selecting beneficiaries, and the required infrastructure.
- **Deploying the initiative:** There are a number of activities to be done once the organization has established an initiative:
  - *Promoting:* In 90% of the organizations that participated in Helix interviews, there were initiatives that were wholly unknown to at least one Helix interviewee. The organization must take an effort to let its employees know of any initiative that they can benefit from. Newer employees who go through some sort of an orientation tend to be more aware of initiatives that they can immediately benefit from. Even those employees who have spent many years in the organization are not very aware of the initiatives that are available to them.

  - *Enabling*: When an employee is interested in a particular initiative and is eligible, the organization must enable the employee to benefit from that initiative. Experiences shared by Helix participants indicate that there are situations when they are unable to take advantage of an organizational initiative since they could not take time off their regular work to participate in a training initiative, or that some procedures diminished the effectiveness of the initiative.

- **Responding to outcomes of initiatives**: When an employee participates and benefits from an initiative, typically, there are new skills or knowledge that are acquired, and the employee could recommend improvements based on this. For example, if an employee receives education or training on systems engineering processes, and if the organization does not support modification of existing systems engineering processes, it defeats the purpose of the education.
- **Evaluating the initiative:** The most critical aspect of the success of an initiative is to evaluate it periodically, and to then update, reform, stop, or restart an organizational initiative. A critical evaluation could also reveal enablers and inhibitors for the initiatives. Helix interviews indicated evidence of many situations:
  - Initiatives no longer address the need for which they were established.
  - The need for which an initiative was established is no longer valid.
  - There are more trainers than trainees.
  - Employees are not motivated.
  - The evaluation of some initiatives makes it appear more successful than it really is.
  - The procedures and policies for an initiative could be burdensome.
  - There is a need to restart an initiative that used to be very effective but was stopped due to many reasons, including budget cuts.
  - The duration of a training course may be altered.
  - The target beneficiaries for an initiative need to be redefined.

8.2.4 Critical Factors for Success with Organizational Initiatives

When individuals discussed successes and failures with organizational initiatives, there were four factors that stood out as critical to the success of any initiative:

- **Establishing the right initiative:** Like in any good systems engineering development, identifying the requirements and addressing them appropriately while establishing the initiative is a necessary first step.
- **Spreading the word:** Any organizational initiatives will be ineffective when an intended beneficiary is unaware that such an initiative exists within the organization. Organizations must take an effort to let its employees know about their eligibility and existence of any organizational initiatives, and enable them to benefit from them.
- **Periodical evaluation of the initiative:** Due to the dynamic nature of the organizational environment, it is important to critically evaluate any initiative periodically to identify modifications that need to be made to the initiative.

- **Commitment from leadership:** Even if many relevant and effective initiatives were available, commitment from the organizational and immediate leadership is essential for an employee to benefit from an initiative.

9 CAREER PATHS

In addition to understanding the overall characterization of the elements of a systems engineer's career, it is helpful to look at the order and overlap of these elements, which can provide additional insights. The output of the *Atlas* method to visualize a career path is illustrated in Figure 13 below.

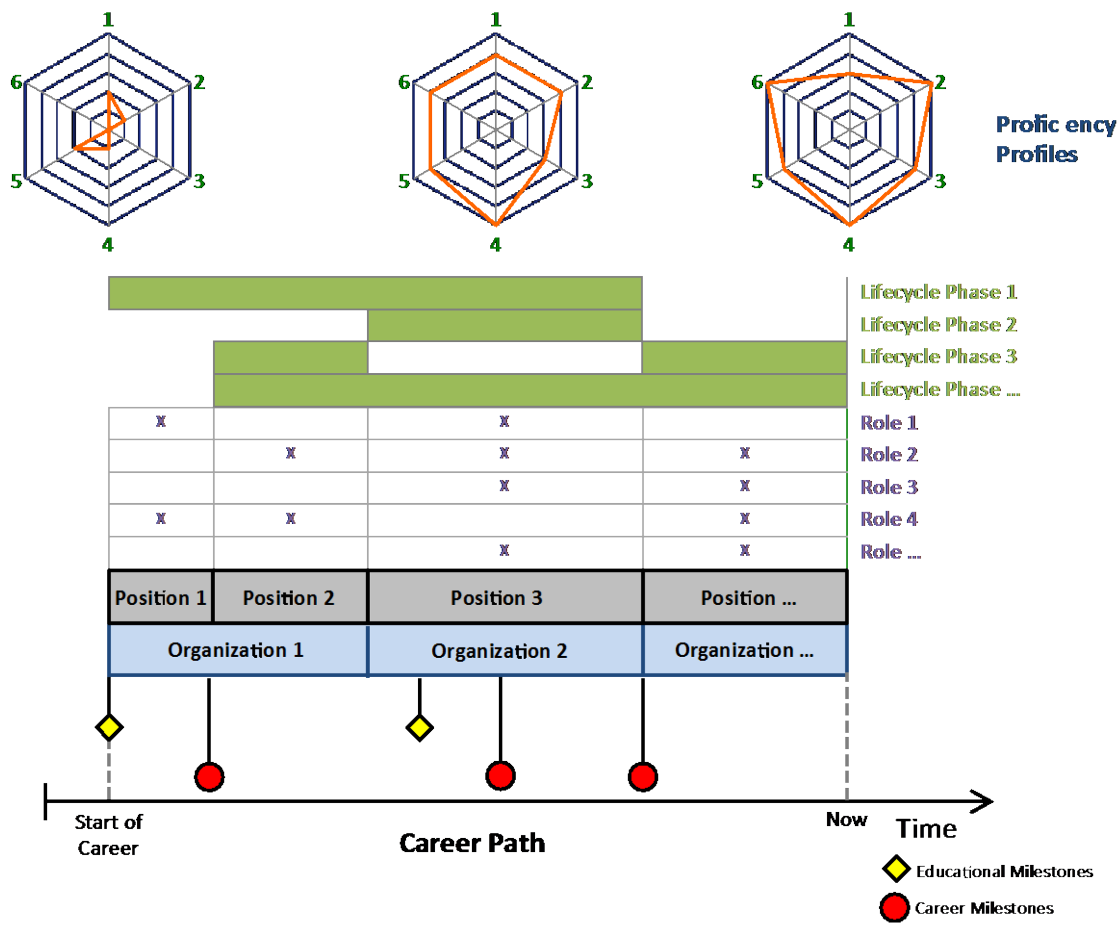


Figure 13. Visualizing a Career Path

The visualization pulls together the following elements of a career path:

- **Timeline:** Time is the dimension onto which all other elements of the career path are projected and visualized. The visualization helps understand the sequence, timing, and duration of various elements of the career path, offering valuable insights for developing the careers of systems engineers.
- **Educational Milestones:** The career of an individual typically begins when an undergraduate degree (or in some cases, a higher degree) is obtained. When, in which disciplines degrees are obtained, and how they impact other elements of the career path, can be observed.
- **Career Milestones:** Significant milestones in terms of types of systems engineering positions,

such as first leadership position, chief systems engineer, or program engineer, etc.

- **Organizations:** The variety of organizations and the time spent in each of those organizations can provide interesting insights, particularly if the organizations vary in terms of sectors, key domains, or other factors.
- **Positions:** The number and duration of all positions held across organizations are captured in the career path. The effect of duration and educational qualifications on positions can be observed.
- **Roles:** The roles performed in each of the above positions perhaps offer the most interesting insights into a career path. An individual is likely to perform more than one role in any particular position, but those roles typically vary as the career progresses. Some roles performed earlier in someone's career may no longer be performed, and there may be newer roles played later in her career. The types of roles performed concurrently offer insights into each position.
- **Lifecycle Phases:** The lifecycle phases experienced during each of the above positions are indicated along the career path. The duration and sequence of the lifecycle experiences indicates the exposure that an individual possesses. Similarly, some roles may be more relevant to particular lifecycle phases.
- **Proficiency Profiles:** The level of proficiency can be profiled at any point in the career. Possibly the most difficult to depict accurately across the career, the proficiency profile can be mapped onto roles and positions an individual performs. This visualization also helps to show how education and experiences influence proficiency.

The career path visualization currently does not include mentoring or training, but gathering all elements into a single visualization provides a holistic view of the entire career of an individual systems engineer. When multiple career paths of different individuals are visualized, patterns can be observed that can offer interesting insights for career development of future systems engineers.

Appendix A provides the basic approach and tools for generating career profiles using paper-based tools. The supplementary Excel based tools (http://www.sercuarc.org/projects/helix/) can be used in lieu of the paper-based tools. The full methodology for career path analysis can be found in the Section 2.5 of the companion Technical Report (SERC-2016-TR-118).

9.1 Implications for Use

Detailed career paths are a method for visualizing how key aspects of an individual's career have changed over time; e.g., the parts of the lifecycle on which an individual worked or the change in proficiency level. As shown in Figure 13, data about the characterization of experiences and education and training are captured along with proficiency assessments over time. These can be a valuable source of insight for an individual – *What steps have I taken or skipped and how has that path impacted my effectiveness?* – or for an organization – *Are there common patterns across the workforce? Where do existing Organizational Development Initiatives impact individuals' career paths?* Organizations considering using career paths for baseline understanding of the various development approaches in their workforce should consider whether there are any key characteristics to add – are there other things the organization would like to capture? For example, key mentoring experiences could be added to the template. To identify workforce level patterns, the key is consistent application – all employees should use the same approach, templates, and criteria to enable effective pattern analysis.

10 *ATLAS* USES AND USE CASES

There are two primary ways in which *Atlas* can be used – to provide insight and guidance to individuals and to inform organizational-level efforts. Guidance on how to use various aspects of *Atlas* is provided throughout the various sections of this document. This section pulls this together, describing at a high level the major expected uses for *Atlas*. Several organizations have, to varying degrees, tried all of them. Figure 14 shows individual uses.

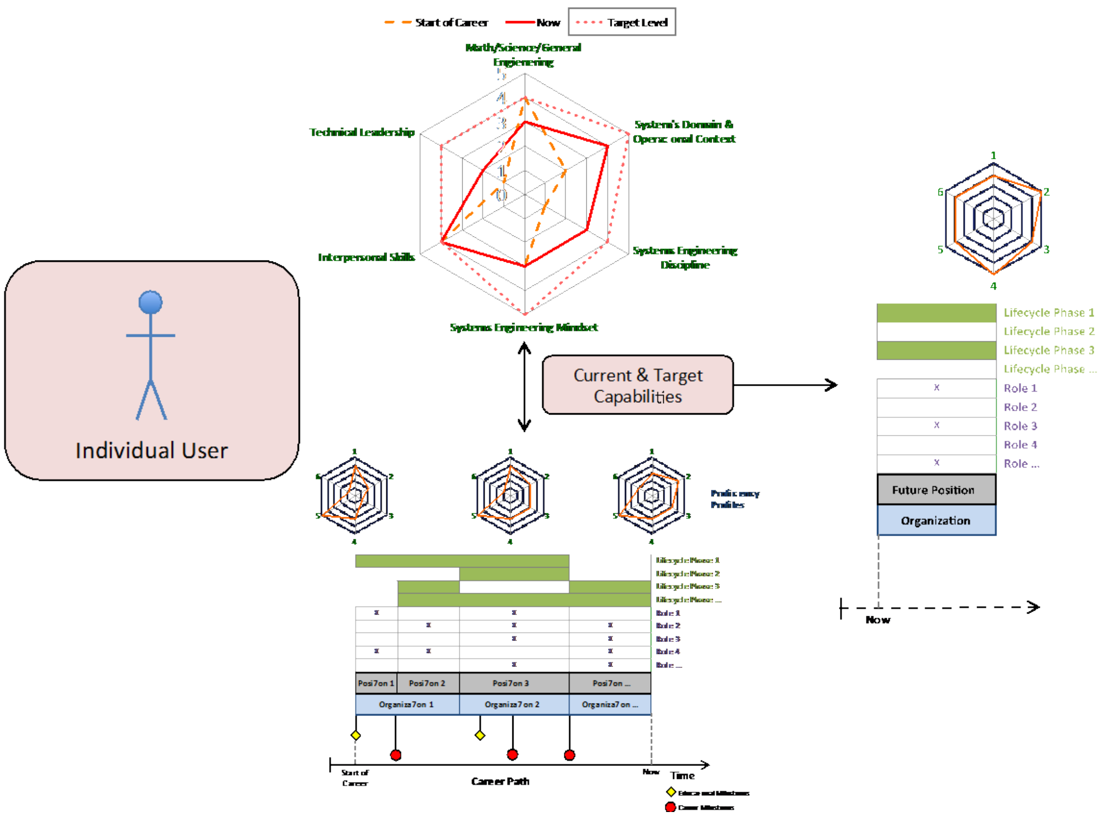


Figure 14. Expected Uses for an Individual User

As shown in Figure 14 an individual is expected to be able to:

1. **Use Proficiency Self-Assessment to identify current proficiency levels as well as past trends**. As described 5.8.1, proficiency profiles are most effective when they are examined over time. An individual will benefit from understanding these patterns and using them to inform potential targets for the future.

2. **Use Career Path self-assessment to categorize and analyze past forces** (experiences, mentoring, and education and training). This data can be used to identify any clear gaps in Forces over time.

3. **Use Proficiency and Career Path self-assessments to identify a way ahead for a career.**

   o **Identify a target state**. Proficiency profiles provide a useful starting point for discussions with the organization about potential future positions – what positions make sense, what the proficiency expectation for this position are, etc. These future goals could be based on known positions within an organization (e.g. "I want to be a systems architect”) or individual desire (e.g. “I am interested in this type of system”). Target states can often be clarified in discussion with a mentor or leader who understands the expectations for different types of positions in the organization as well as the individual’s proficiencies.

- **Assess gaps between current and target proficiency.** As illustrated in Section 5.8.1., once target proficiencies have been identified, they can be plotted in a proficiency profile along with current proficiency levels. This provides an easy way to visualize gaps between current and target proficiency, helping an individual understand where they need to focus their growth.
- **Pair proficiency gaps with career path information to identify potential ways to improve proficiency.** Experiences, mentoring, education, or training are all ways that proficiencies can be improved and often a combination of forces is required to reach a target proficiency. For example, a gap in systems engineering discipline may initially be addressed by targeted training or education programs. However, a best practice identified by Helix is that this must be applied on the job immediately in order for any improvements in proficiency to become permanent. If a mentor can help guide the application of new learning in these experiences, there is likely to be additional improvement in proficiency as well. All of these considerations provide a starting point for planning and can be used to discuss possibilities with management or leadership.

Figure 15 shows expected organizational uses of *Atlas*.

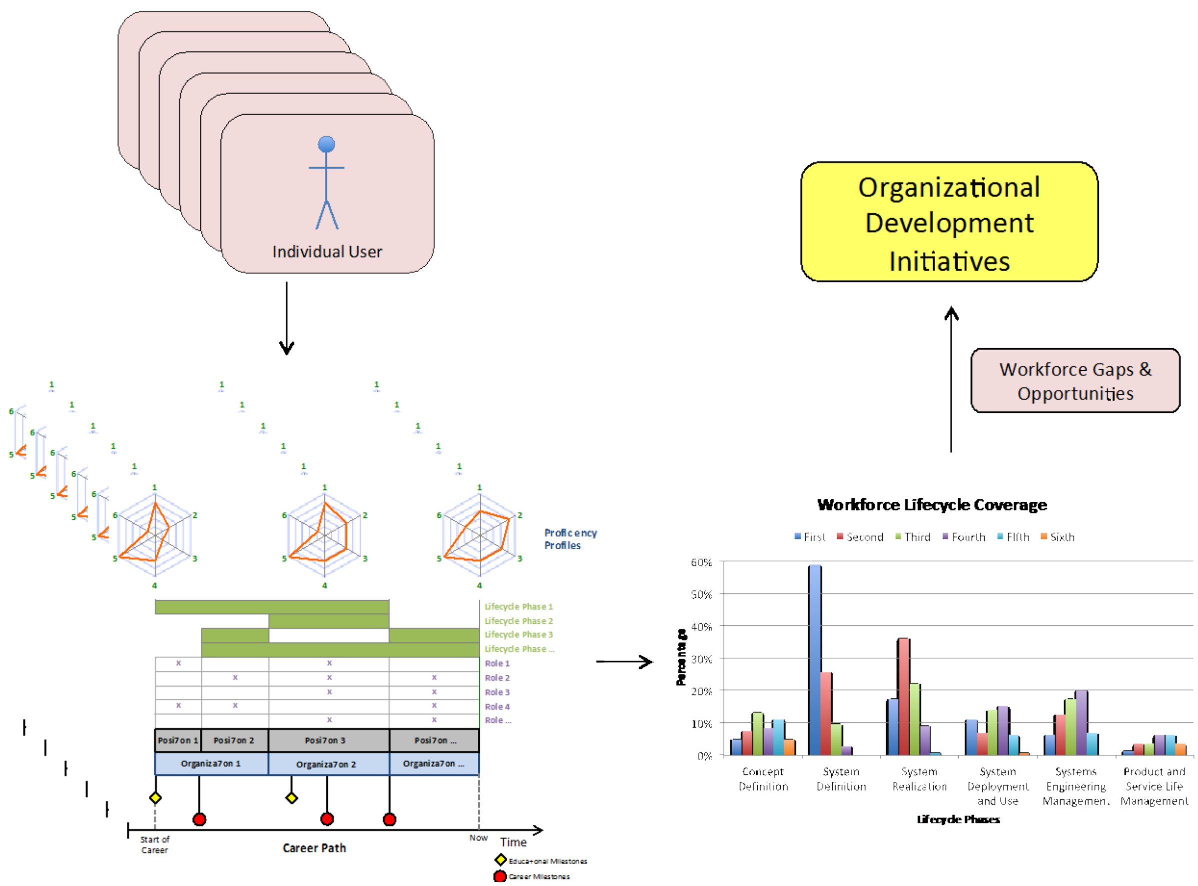


Figure 15. Expected Uses for Organizations

As shown in Figure 15, an organization is expected to be able to:

- **Treat the workforce as a collection of individuals.** Each individual can gain insight on current and potential target capabilities as discussed above. By taking the proficiency profiles – current and target – for a group of individuals, the organization can gain insight into any current capability gaps and understand desired future capabilities. For example, if no one in the group has higher than a proficiency level "6" in Technical Leadership, but the organization feels it needs several individuals with a level "8" proficiency or higher, then the organization has identified a critical skills gap. Paired with the target states, the organization can then identified individuals who are already interested in developing their Technical Leadership skills and can focus opportunities related to technical leadership on these individuals. Likewise, the organization may identify individuals who are believed to be "high potential" for technical leadership who may not have identified this in themselves and enable a conversation about future directions.

  The Helix team recognizes that there is likely to be some systemic changes from viewing the workforce holistically, rather than as a collection of individuals – "the whole is greater than the sum of its parts" – but the research to date has not enabled the team to understand this. Future research will include modeling to support holistic workforce-level analysis.

- **Use the career path data from individuals to identify patterns of the overall workforce.** Similar to the point above, organizations can use the career path data for the individuals in the workforce to identify overall patterns. For example, perhaps less than 5% of the workforce has experience in the role of "Concept Creator". If the organization has identified this as a critical area for growth of systems engineers, this may indicate that the organization should develop initiatives to foster growth in this area. Likewise, if there is an area of the lifecycle that is commonly missed in the workforce, the organization can determine if this is a critical gap or whether it makes sense in the organizational context. For example, if only 10% of the workforce has experiences in "Systems Deployment and Use", but the organization does not participate in operation and maintenance of its systems, then this may be seen as acceptable. The organization also now has data about the workforce that it can use to fill gaps. For example, if the organization needed perspective on a project specific to "Systems Deployment and Use", the data will provide insight on who in the organization has this experience.

- Use workforce data to improve or create new organizational development initiatives. Using the gap analysis across current and future desired capabilities, the organization can identify opportunities or set strategic goals regarding workforce capability. As illustrated in the examples above, this information would then provide opportunities for improved or new development initiatives.

REFERENCES

Bellinger, G. 2004. "Modeling and Simulation: An Introduction." Webpage. Accessed 30 August 2014. Available at: http://www.systems-thinking.org/modsim/modsim.htm

BKCASE Editorial Board. 2016. *The Guide to the Systems Engineering Body of Knowledge (SEBoK), v. 1.7*. R.D. Adcock (EIC). Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed 27 October 2016. www.sebokwiki.org.

Cameron, K. and R.E. Quinn. 2011. *Diagnosing and Changing Organizational Culture: Based on the Competing Values Framework*: Based on the Competing Values Framework, 3rd Edition. Hoboken, NJ: Wiley.

Coldstream, P. 2006. "Training Minds for Tomorrow's World: on Independence, Curiosity, and Effectiveness." *Higher Education in Europe*. 22(4): 507-515.

Harris, G. and K. Eikenberry. 2016. "DISC Profile and Human Behavioral Assessments Provider: What is the DISC Model?" https://discpersonalitytesting.com/blog/what-is-the-disc-model/

DoD. 2013. *Interim DoD Instruction 5000.2: Operation of the Defense Acquisition System*. Arlington, VA: U.S. Department of Defense. 26 November 2013.

Freshwater, D. (ed.). 2002. *Therapeutic Nursing: Improving Patient Care through Self Awareness and Reflection.* London, England: Sage Publications Ltd.

Harrison, R. 1972. "Understanding your organisation's character." Harvard Business Review.

Hutchison, N. *A Framework to Classify Experiences and Enable Career Path Analysis to Support Maturation of Effective Systems Engineers in the Defense Industry*. PhD Dissertation. Hoboken, NJ: Stevens Institute of Technology. October 2015.

Hutchison, N., A Pyster, D. Henry. 2015. "Trends in the Education of Systems Engineers." Submitted to *Systems Engineering* March 2015.

Hutchison, N., A. Pyster, D. Henry, C. Barboza. 2015. "Understanding What Makes Systems Engineers Effective." Submitted to *Systems Engineering* March 2015.

INCOSE. 2014. *A World in Motion: Systems Engineering Vision 2025*. San Diego, CA, USA: International Council on Systems Engineering.

ISO/IEC. 2009. "Systems and Software Engineering Vocabulary (SEVocab)" - ISO/IEC 24765. in International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) [database online]. Geneva, Switzerland, 2009 [cited December 21 2009]. Available at: http://pascal.computer.org/sev_display/index.action.

ISO/IEC/IEEE. 2008. *Systems and Software Engineering -- System Lifecycle Processes.* Geneva, Switzerland: International Organisation for Standardisation/International Electrotechnical Commissions. ISO/IEC/IEEE 15288:2008.

ISO/IEC/IEEE. 2010. *Systems and Software Engineering - System and Software Engineering Vocabulary (SEVocab)*. Geneva, Switzerland: International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)/ Institute of Electrical and Electronics Engineers (IEEE). ISO/IEC/IEEE 24765:2010.

Koen, J., U-C. Klehe, and A.E.M Van Vianen. 2012. “Training Career Adaptability to Facilitate a Successful School-to-Work Transition.” *Journal of Vocational Behavior*. 81(3): 395-408. December 2012.

Maximini, D. 2015. *The Scrum Culture: Introducing Agile Methods in Organizations*. Cham, Switzerland: Springer.

Meyers and Briggs Foundation, The. 2016. “The Meyers-Briggs Type Indicator: MBTI Basics.” Gainesville, FL: The Meyers & Briggs Foundation. http://www.myersbriggs.org/my-mbti-personality-type/mbti-basics/

Pyster, A., D.H. Olwell, T.L.J. Ferris, N. Hutchison, S. Enck, J. Anthony, and D. Henry (eds.). 2015. *Graduate Reference Curriculum for Systems Engineering (GRCSE®) v.1.1*. Hoboken, NJ, USA: Trustees of the Stevens Institute of Technology.

Pyster, A., P. Dominick, D. Henry, N. Hutchison, C. Lipizzi, M. Kamil, S. Manchanda. 2014(b). Atlas*: The Theory of Effective Systems Engineers*, v. 0.25. Hoboken, NJ: Systems Engineering Research Center, Stevens Institute of Technology. SERC-2014-TR-038-4.

Pyster, A., R.L. Pineda, D. Henry, N. Hutchison. 2013. *Helix – Phases 1 and 2.* Hoboken, NJ: Systems Engineering Research Center, Stevens Institute of Technology. SERC-2013-TR-038-2.

Pyster, A., R.L. Pineda, D. Henry, N. Hutchison. 2014. *Helix – Phase 3*. Hoboken, NJ: Systems Engineering Research Center, Stevens Institute of Technology. SERC-2014-TR-038-3.

Sheard, S. 1996. “Twelve Systems Engineering Roles.” Proceedings of the International Council on Systems Engineering (INCOSE) Sixth Annual International Symposium, Boston, Massachusetts, USA, 1996.

Sheard, S. 2000. “Systems Engineering Roles Revisited.” Software Productivity Consortium, NFP.

Glossary and Terminology

Consistency in the definition and understanding of terminology and concepts is essential for any deliberation. This section presents the definitions and classifications that are relevant to *Atlas*. Some have been obtained from available literature, while others have been created specifically for *Atlas*.

Acronyms and Abbreviations

CSE	Chief Systems Engineer
DASD(SE)	U.S. Deputy Assistant Secretary of Defense for Systems Engineering
DIB	Defense Industrial Base (that supports DoD)
DoD	U.S. Department of Defense
GRCSE	Graduate Reference Curriculum for Systems Engineering
HR	Human Resources
INCOSE	International Council on Systems Engineering
IPT	Integrated Product Team
IR&D	Internal (or Independent) Research & Development
IRB	Internal Review Board
IT	Information Technology
IV&V	Integration, (or Independent) Verification, & Validation
MBA	Master of Business Administration
NDIA-SED	National Defense Industrial Association – Systems Engineering Division
PEO	Program Executive Office
PLM	Product Life Management
QRC	Quick Reaction Capability
SE	Systems Engineering
SERC	Systems Engineering Research Center
SEBoK	*Guide to the Systems Engineering Body of Knowledge*
SME	Subject Matter Expert
SPRDE	Systems Planning, Research, Development, and Engineering
UARC	University-Affiliated Research Center
V&V	Verification & Validation
PM	Project (or Program) Manager

Atlas Definitions

- **Systems Engineer**

> *A* ***Systems Enginee****r is an individual who performs systems engineering activities and is recognized (either formally or informally) by his or her organization for her ability to perform these activities.*

This definition of a systems engineer does *not* refer to the title that someone may hold in her organization. Someone may never hold the title 'Systems Engineer', but could be considered to be one based on the activities she performs. Similarly, someone may hold the title 'Systems Engineer', but her activities may not be considered to be systems engineering activities.

- **Effective Systems Engineer**

> *An **Effective Systems Engineer** is someone who consistently delivers value by performing systems engineering activities in positions assigned by the organization.*

This definition is fundamental to *Atlas* since the focus of Helix research is the effectiveness of systems engineers. Though 'effectiveness' is a subjective term, this definition ties it to 'value' that can be defined and even measured – qualitatively, if not quantitatively.

- **Chief Systems Engineer (CSE)**

> *A **Chief Systems Engineer (CSE)** is one who has formal responsibility to oversee and shepherd the technical correctness and to maintain a consistent vision for a system, often coordinating with many other systems engineers who have smaller scopes of responsibility.*

The Chief Systems Engineer (CSE) position is one of the most senior technical positions that system engineers can achieve while staying in a technical track (as opposed to a management track). Though the title 'Chief Systems Engineer' is not used in all organizations, the concept of a CSE position (or equivalent) is common, especially in industry. There is no consistent description of a CSE's (or equivalent's) formal authority, but overall responsibility for a system is often split in some way between the CSE and the project or program manager (PM).

- **Position**

> *A **Position** held by an individual is equivalent to a 'title', where the organization defines what roles and responsibilities it entails.*

This definition of a position is usually specific to an organization and does not translate across organizations.

- **Role**

> *A **Role** performed by an individual consists of a specific set of related activities.*

Typically, an individual performs multiple roles in any given position. In the context of *Atlas*, the roles of interest are systems engineering roles.

- **Career Path**

> *An individual's **Career Path** is the precise combination (in terms of characteristics, timing, and order) of experiences, mentoring, and education and training that they undergo during their entire career.*

This definition, created for *Atlas*, is different from how career paths are typically defined in the human resources (HR) community. HR definitions tend to be focused on rigid hierarchy that may be useful for HR classification and management of positions within an organization. However, they provide little insight into the growth and development of individuals throughout their career, particularly across organizations.

- **Proficiency**

  *The **Proficiency** of an individual is the quality or state of knowledge, skills, abilities, behaviors, and cognition.*

  In *Atlas*, the term 'proficiency' is used broadly to include everything that an individual needs to be good at in order to be an effective systems engineer. This distinguishes *Atlas* from competency models that tend to focus primarily on the discipline of systems engineering.

ATLAS CLASSIFICATIONS

- **Seniority of a Systems Engineer**

  As systems engineers traverse the path of their careers from the point of entry into the workforce (or recruitment) to the point or exit from the workforce (or retirement), there is a continual maturation that is reflected in the breadth and depth of their proficiencies; the types of roles & positions they play; and the value that they provide or that is expected from them. Grouping systems engineers under some levels of 'seniority' that reflect the levels of maturation enables patterns to be identified across systems engineers, and insights to be drawn from them.

  Helix has identified three levels of seniority in systems engineers: junior, mid-level, and senior. Traditionally, 'number of years of work experience' has been used as a preliminary criterion for distinguishing between these levels of seniority, but it fails to capture the nuances of differentiation within systems engineers. Hence, it is not included in Table 6 that states various criteria used to distinguish between junior, mid-level, and senior systems engineers. These criteria are meant to be indicative and not rigid; there are always examples of specific individuals whose seniority is not consistent with these criteria.

Table 6. Criteria for Distinguishing the Seniority of Systems Engineers

	Junior	Mid-level	Senior
1.	Not more than 1 formal leadership position	At least 2 formal leadership positions	More than 2 formal leadership positions
2.	Experiences primarily in components	Experiences in components and subsystems, and perhaps in systems	Experiences in components, subsystems, systems, and perhaps in systems of systems
3.	Experiences in at least 2 aspects of the systems lifecycle	Experiences in at least 3 aspects of the systems lifecycle	Experiences in at least 4 aspects of the systems lifecycle

With respect to Table 6:

1. Experience is considered to be 'relevant' if it directly supports the growth of systems engineering proficiencies.
2. A leadership position is 'formal' if it is officially defined and recognized by the organization. This does not mean that the individual necessarily has organizational

authority over the individuals she is leading. Likewise, there is no defined minimal team size. Typically, early leadership positions are over small teams (less than five people) and as the individual matures, the size of the teams increases.

3. The hierarchy of system levels (components -> subsystems -> systems -> system of systems) is based on definitions from the *Guide to the Systems Engineering Body of Knowledge* (BKCASE Editorial Board 2016) and reflects system complexity and completeness, where ‘parts’ at any level are combined to form the ‘whole’ at the next level.

4. The various aspects of the systems lifecycle are based on definitions from the *Guide to the Systems Engineering Body of Knowledge* (BKCASE Editorial Board 2016) and are elaborated in Section 5.4.

5. Formal education, titles, and roles are *not* considered to be distinguishing criteria, since they cannot be used to consistently draw any distinctions between levels of seniority of systems engineers. However, as a baseline, systems engineers typically have an undergraduate degree in a STEM (science, technology, engineering, and mathematics) field.

APPENDIX A: SELF-ASSESSMENT TOOLS (PAPER BASED)

This appendix provides the paper-based tools for assessment generated by the Helix team. These tools are easy and simple for an individual to use to gain insight into his or her career or for an organization to deploy to enable career planning with its employees. The materials include the templates and some basic guidance on how to use them. However, these materials do not include the depth of detail included in the Excel-based templates. Particularly if an organization intends to collect data from many of its employees based on *Atlas*, it is recommended that the Excel-based files be used. They are available at http://www.sercuarc.org/projects/helix/ under "Deliverables."

The content of the tools is outlined below:

Note that there is also an Excel-based tool. This tool, and a brief user guide, are available at the Helix website (http://www.sercuarc.org/projects/helix/).

Instructions for Completing a Proficiency Self-Assessment

Overview

Proficiency defines the knowledge, skills, abilities, behaviors, patterns of thinking, and abilities that are critical to the effectiveness of systems engineers. The *Atlas* proficiency model consists of six difference proficiency areas:

- **Math/Science/General Engineering**: Foundational concepts from mathematics, physical sciences, and general engineering;
- **System's Domain & Operational Context**: Relevant domains, disciplines, and technologies for a given system and its operation;
- **Systems Engineering Discipline**: Foundation of systems science and systems engineering knowledge;
- **Systems Engineering Mindset**: Skills, behaviors, and cognition associated with being a systems engineer;
- **Interpersonal Skills**: Skills and behaviors associated with the ability to work effectively in a team environment and to coordinate across the problem domain and solution domain; and
- **Technical Leadership**: Skills and behaviors associated with the ability to guide a diverse team of experts toward a specific technical goal.

Each of these areas contains several categories, or groupings of related knowledge, skills, abilities, behaviors, or cognitions, as illustrated in Table 1.

Self-Assessment

In order to perform a self-assessment, individuals are asked to review the definitions of the proficiency areas above and the categories in Table 1. Additional detail can be found in the full report on *Atlas 1.0*, SERC-2016-TR-118, found at the Helix webpage (http://www.sercuarc.org/projects/helix/). Then use the template to generate a "0 to 10" initial assessment of your current proficiency in each Area, with "0" meaning you have no skill in the area and 10 meaning your skills are the top within your experiences. Consider the following guidelines:

- For each Proficiency Area, think about proficiency across *all categories*, not just one. For example, if you are a "10" in a single category, but a "5" in all others, you would not be a "10" for the entire Area.
- For each Area, think about what is most critical for your current position. This may not change your assessment, but may mean that a lower number not an issue.
- Consider your past experiences in the Area, any training or education that might be relevant, and where you might have received guidance from a mentor or leader. These things together will have shaped your proficiency, and thinking about them may help you to assess yourself more realistically.
- You know your strengths and areas for growth – be honest in your responses.

A proficiency rubric for further guidance can be found on page 78.

Once you have completed your initial assessment for your *current* proficiency, you can choose to retroactively assess what your proficiency was at different points in your career. For example, when you completed your undergraduate education or joined your current organization. This may help you to better reflect on changes over time. If you do this, revisit your current proficiency assessment afterwards and determine whether any adjustments are required.

Area	Category	Topic
1. Math / Science / General Engineering	1.1. Natural Science Foundations	
	1.2. Engineering Fundamentals	
	1.3. Probability and Statistics	
	1.4. Calculus and Analytical Geometry	
	1.5. Computing Fundamentals	
2. Systems' Domain & Operational Context	2.1. Principal and Relevant Systems	< List of Principal and Relevant Systems >
	2.2. Familiarity with Principal System's Concept of Operations (ConOps)	
	2.3. Relevant Domains	< List of relevant Domains >
	2.4. Relevant Technologies	< List of relevant Technologies >
	2.5. Relevant Disciplines and Specialties	< List of relevant Disciplines and Specialties >
	2.6. System Characteristics	< List of applicable System Types, Scales, and Levels >
3. Systems Engineering Discipline	3.1. Lifecycle	3.1.1 Lifecycle Models 3.1.2 Concept Definition 3.1.3 System Definition 3.1.4 System Realization 3.1.5 System Deployment and Use 3.1.6 Product and Service Life Management
	3.2. Systems Engineering Management	3.2.1 Planning 3.2.2 Risk Management 3.2.3 Configuration Management 3.2.4 Assessment and Control 3.2.5 Quality Management
	3.3. SE Methods, Processes, and Tools	3.3.1 Balance and Optimization 3.3.2 Modeling and Simulation 3.3.3 Development Process 3.3.4 Systems Engineering Tools
	3.4. Systems Engineering Trends	3.4.1 Complexity 3.4.2 Model Oriented Systems Engineering 3.4.3 Systems Engineering Analytics 3.4.4 Agile Systems Engineering
4. Systems Engineering Mindset	4.1. Big-Picture Thinking	
	4.2. Paradoxical Mindset	4.2.1 Big-Picture Thinking and Attention to Detail 4.2.2 Strategic and Tactical 4.2.3 Analytic and Synthetic 4.2.4 Courageous and Humble 4.2.5 Methodical and Creative
	4.3. Flexible Comfort Zone	
	4.4. Abstraction	
	4.5. Foresight and Vision	
5. Interpersonal Skills	5.1. Communication	5.1.1 Audience 5.1.2 Content 5.1.3 Mode
	5.2. Listening and Comprehension	
	5.3. Working in a Team	

Area	Category	Topic
	5.4. Influence, Persuasion and Negotiation	
	5.5. Building a Social Network	
6. Technical Leadership	6.1. Building and Orchestrating a Diverse Team	
	6.2. Balanced Decision Making & Rational Risk Taking	
	6.3. Guiding Stakeholders with Diverse/Conflicting Needs	
	6.4. Conflict Resolution & Barrier Breaking	
	6.5. Business and Project Management Skills	
	6.6. Establishing Technical Strategies	
	6.7. Enabling Broad Portfolio-Level Outcomes	

Proficiency Self-Assessment Rubric

Atlas Proficiency Area / Category	Proficiency Level "1"	Proficiency Level "3"	Proficiency Level "5"
1. Math / Science / General Engineering			
1.1. Natural Science Foundations	Minimal awareness of the basic concepts of physics, chemistry, and biology		Expert in the principles and concepts of physics, chemistry and biology including practical experience, and ability to apply these in the system's context
1.2. Engineering Fundamentals	Minimal awareness of the basic engineering concepts, processes, and techniques.		Expert in basic engineering concepts, processes, and techniques including practical experience, and ability to apply these in the system's context
1.3. Probability & Statistics	Minimal awareness of the basics of probability and statistics		Expert in probability theory, probability distributions, statistical measures and other related topics, and ability to readily apply them where required
1.4. Calculus & Analytical Geometry	Minimal awareness of differential calculus, integral calculus, coordinate systems, and geometric equations		Expert in differential and integral calculus methods, coordinate systems, transformations, describing and analyzing geometric objects and ability to readily apply them where required
1.5. Computing Fundamentals	Minimal awareness of computer organization, operating systems, and programming languages		Expert in computer architectures, networking, operating systems, programming languages and ability to readily apply them where required
2. Systems' Domain & Operational Context			
2.1. Principal and Relevant Systems	Minimal knowledge about the specific systems		Expert in the specific systems, their development and operation
2.2. Familiarity with System's Concept of Operations (ConOps)	Minimal awareness of the ConOps of the principal system		Expert in the ConOps of the system, and the ability to comprehensively develop ConOps

Atlas Proficiency Area / Category	Proficiency Level "1"	Proficiency Level "3"	Proficiency Level "5"
2.3. Relevant Domains	Minimal familiarity with the terminology and basic concepts of the specific domains		Expert in the domain and the development and operation of systems in that domain.
2.4. Relevant Technologies	Minimal familiarity with the terminology and basic concepts of the specific technologies		Expert in the technology and its current development, and the ability to easily apply it to system development
2.5. Relevant Disciplines and Specialties	Minimal familiarity with the terminology and basic concepts of the specific disciplines		Expert in the discipline and latest advancements
2.6. System Characteristics	Minimal familiarity with the specific Types, Scales, and Levels of systems		Expert in the specific Types, Scales, and Levels of systems
3. Systems Engineering Discipline			
3.1. Lifecycle	Minimal awareness of lifecycle models and lifecycle stages		Expert in the understanding of lifecycle models and how systems are developed in them. A deep understanding of specific lifecycle stages of system development and ability to carry out the required technical activities at those stages
3.2. Systems Engineering Management	Minimal awareness of systems engineering management activities		Expert in specific topics of systems engineering management and ability to perform the required management activities
3.3. SE Methods, Processes, & Tools	Minimal awareness of SE methods, processes and tool in an isolated manner		Expert in specific SE methods, processes, and tools, and in the application of these.
3.4. Systems Engineering trends	Minimal awareness of the specific trends and their application to systems development		Expert in the specific trends and their application to systems development

Atlas Proficiency Area / Category	Proficiency Level "1"	Proficiency Level "3"	Proficiency Level "5"
4. Systems Engineering Mindset			
4.1. Big-Picture Thinking	Minimal ability to think beyond a narrow scope of the problem at hand	Able to think in a limited manner outside a narrow scope with some guidance	Expert in thinking broadly along various dimensions (e.g., regarding broader domain or enterprise-level considerations, and linking across apparent disparate domains such as incorporating "soft" science with "hard" science)
4.2. Paradoxical Mindset	Minimal ability to handle seemingly opposed views	Able to understand the one of the opposed views separately but not both	Expert in the understanding of two opposed views and perspectives, ability to successfully handle them both separately and together, and the ability to successfully move from one perspective to another
4.3. Flexible Comfort Zone	Comfortable only strictly within one's comfort zone and area of technical expertise	Able to permeate beyond one's comfort zone in a limited manner, but hesitates to explore the unknown	Willing and able to permeate the boundaries of one's comfort zone with ease, and able to comfortably explore the unknown and readily seek interdisciplinary SME
4.4. Multi Scale Abstraction	Minimal ability to abstract or infer from individual pieces of information and relate to environmental context	Able to abstract insights with some guidance and prior experience and understand system in larger operational context	Expert in quickly and effectively abstracting (from highly detailed level to highly conceptual level) new and significant insights from seemingly disparate pieces of information across system and environmental scales
4.5. Foresight & Vision	Minimal ability to comprehend future impacts of current decisions and situations	Able to comprehend impacts in the near future, in a limited manner	Expert in seeing future impacts of current decisions, and to clearly visualize future stages of a system's lifecycle

Atlas Proficiency Area / Category	Proficiency Level "1"	Proficiency Level "3"	Proficiency Level "5"
5. Interpersonal Skills			
5.1. Communication	Minimal ability to successfully communicate any information to any audience in any mode	Able to communicate well in one predominant mode with limited familiar audience	Expert in being able to successfully and unambiguously communicate to a variety of audience and a wide range of technical and non-technical content, in various written and oral modes.
5.2. Listening & Comprehension	Minimal ability to listen to and understand others' points and perspectives	Able to listen to other's points, but limited ability to comprehend	Expert in listening and successfully comprehending others' points and perspectives
5.3. Working in a Team	Minimal ability to work with anyone else, preferring to work alone	Able to work in familiar teams, but limited ability to work on new teams	Very comfortable to work with others, and being able to quickly and successfully become part of any team exhibiting positive team dynamics
5.4. Influence, Persuasion & Negotiation	Minimal ability to modify another person's viewpoint or perspective, even when that is detrimental	Able to influence others in a limited manner, only with familiar individuals or when they are not experts in their own fields	Expert in positively influencing others, particularly experts in their own fields, to see beyond their viewpoints and to come to agreements for the good of the overall system
5.5. Building a Social Network	Minimal ability to form any social relationship with a professional acquaintance	Able to form a limited social network among those with frequent interactions	Expert in establishing strong social relationships with professional acquaintances both within and outside the organization
6. Technical Leadership			
6.1. Building & Orchestrating a Diverse Team	Minimal ability to form or lead a team with any success	Able to build a team with guidance but has difficulty in handling or delegating to a diverse team	Expert in bringing together the right team for the task, being able to synergistically draw individual strengths of team members, successfully leading the team to achieve end goal

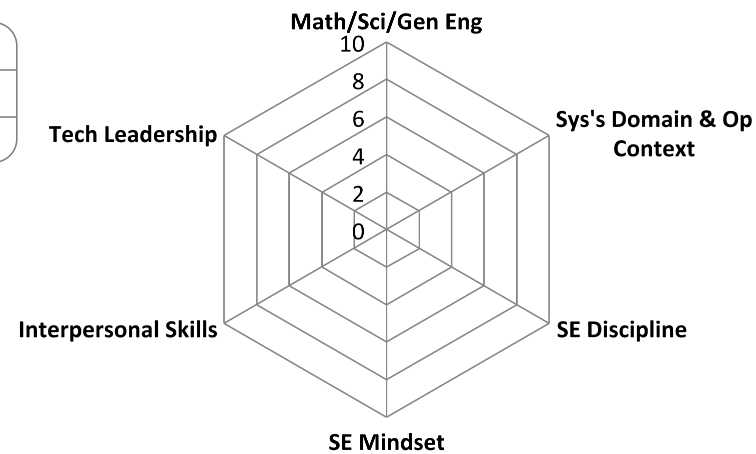
Atlas Proficiency Area / Category	Proficiency Level "1"	Proficiency Level "3"	Proficiency Level "5"
6.2. Balanced Decision Making & Rational Risk Taking	Minimal ability to take balanced decisions or to take any rational risks	Able to take some balanced decisions with some guidance, but limited ability to take rational risks	Expert in taking successful decision considering all relevant factors and constraints, and being able to rationally calculate risks when required
6.3. Guiding Stakeholders with Diverse/ Conflicting Needs	Minimal ability to guide internal and external stakeholders and their needs	Able to guide familiar stakeholders, who have well established needs, in a limited manner	Expert in leveraging good relationships with internal and external stakeholder, and successfully meeting their needs
6.4. Conflict Resolution & Barrier Breaking	Minimal ability to resolve any conflict that negatively affects the system, and unable to break barriers of opinions and perspectives that prevent any progress	Able to resolve minor conflict mostly among familiar individuals	Expert in successfully resolving conflict between individuals or teams for the sake of the overall system, and able to break down various technical or cultural barriers
6.5. Business & Project Management Skills	Minimal ability to perform business and project management activities	Able to perform business and project management activities with some guidance and reference	Expert in the knowledge, understanding, and application of various business and project management skills.
6.6. Establishing Technical Strategies	Tactical approach to technology on a project-by-project basis		Develops technical strategies that impact multiple projects (e.g. investment decisions, prioritization of technology roadmaps, etc.)
6.7. Enabling Broad Portfolio-Level Outcomes	Focuses only on outcomes for individual projects		Identifies issues and opportunities that impact an entire portfolio of systems Communicates these issues to leadership and engineers

Proficiency Self-Assessment Tool

Target State:	

ay:

Target State:



Instructions for Completing a Career Path Assessment

Overview

An individual's career path is the precise combination of experiences, mentoring, education, and training that an individual goes, particularly their characteristics, timing, and order. In order to complete a career assessment, an individual should work through the steps outlined here while filling out the career path template.

Experiences

The Helix team chose to use a **position** as the unit of measure for experiences; a position is established by the organization and defines the roles and responsibilities to be performed.

Based on both the literature and the Helix data itself, each position has several characteristics:

- **Relevance:** A 'relevant' position is one that enables a systems engineer to develop the proficiencies critical to systems engineering. Determine a starting point for relevant experiences; this will become the first position (P1) of the career path. Fill in the title and the year(s) for the position(s).
- **Organizations:** Fill out the name of the organization for each position. This will help to show any transition or variation between organizations.
- **Roles:** A role is a collection of related systems engineering activities. Roles were identified based on the activities consistently performed by systems engineers. There are 16 roles identified in *Atlas*, as described in Table 1, below. For each position, review your activities and responsibilities and write down *all* roles played during that position.
- **Lifecycle Phases:** Generic systems engineering lifecycle phases considered in *Atlas* are based on the lifecycle phases in the *Guide to the Systems Engineering Body of Knowledge (SEBoK*), as explained on page 5. (BKCASE Authors 2016) For each position, fill in the area(s) of the lifecycle you worked on.
- **Key Milestones.** Note any key changes in types of positions under key milestones. For example, first systems engineering role, first chief systems engineer role, first supervisory position, etc. would all be indicators of change or growth over career.

Education and Training

Note any educational milestones or key training milestones with the position/timeline in which they occurred. Education milestones may include the completion of a degree or participation in a course that was particularly relevant or impactful for your career. Key training is training that was particularly impactful or useful for your career. You do not need to include training that did not have an impact.

Other

Your organization may ask you to add other information, such as participation in professional societies, publications, etc. to your career path.

Role Name	Role Description
Roles Focused on Systems	
Concept Creator	Individual who holistically explores the problem or opportunity space and develops the overarching vision for a system(s) that can address this space. A major gap pointed out to the Helix team – particularly when working to implement the findings of Helix – has been that of the development of an overarching system vision. This is a critical first step in the systems lifecycle, and several organizations stated that they believed it needed to be separated out. In addition, when looking to the future of what systems engineers need to do (e.g. INCOSE Vision 2025 (2015)), the focus on early engagement and setting the vision was deemed critical.
Requirements Owner	Individual who is responsible for translating customer requirements to system or sub-system requirements; or for developing the *functional* architecture. This is unchanged from (Sheard 1996).
System Architect	Individual who owns or is responsible for the architecture of the system. This is an update of Sheard's "System Designer" role (1996). There was concern both at community events and during later interviews that nowhere in the presented framework did the critical role of systems engineers in architecture come out clearly. Some also argued that "Design" gave the impression that this roles focuses specifically on the details of systems design over architecture.
System Integrator	Individual who provides a holistic perspective of the system; this may be the 'technical conscience' or 'seeker of issues that fall in the cracks' – particularly, someone who is concerned with interfaces. Likewise, there was concern over the word "Glue", which many expressed was not clearly descriptive enough.
System Analyst	Individual who provides modeling or analysis support to system development activities, and helps to ensure that the system as designed meets he specification. This is unchanged from Sheard's roles (1996).
Detailed Designer	Individual who provides technical designs that match the system architecture; an individual contributor in any engineering discipline who provides part of the design for the overall system. This is an addition based on the Helix data. While systems engineers do not always get involved with detailed design, in smaller organizations or on smaller projects it is more common. Likewise, systems engineers who had played this role explained that it was critical in developing their own technical and domain expertise as well as in understanding the design approaches of classic engineers.
V&V Engineer	Individual who plans, conducts, or oversees verification and validation activities such as testing, demonstration, and simulation. This is unchanged from Sheard's roles (1996).
Support Engineer	Individual who performs the 'back end' of the systems lifecycle, who may operate the system, provide support during operation, provide guidance on maintenance, or help with disposal. This was previously titled "Logistics and Operations Engineer" in Sheard (1996). However, in interviews and at community events, the Helix team received feedback that using this title gave the impression that this role was limited and did not encompass the full spectrum of systems engineers' activities at system deployment or post-deployment. Likewise, in several organizations, "logistics" and "operations" were seen as separate disciplines from systems engineering, which caused some contention in discussions. The renaming of this category is intended to address these issues.

Roles Focused on SE Process and Organization	
Systems Engineering Champion	Individual who promotes the value of systems engineering to individuals outside of the SE community - to project managers, other engineers, or management. This may happen at the strategic level or could involve looking for areas where systems activities can provide a direct or immediate benefit on existing projects. Sheard recommended that a role such as this, labeled in her work, as "Systems Engineering Evangelist", be added in (2000).
Process Engineer	Individual who defines and maintains the systems engineering processes as a whole and who also likely has direct ties into the business. This individual provides critical guidance on how systems engineering should be conducted within an organization context. This is unchanged from Sheard's roles (1996).
Roles Focused on Teams	
Customer Interface	Individual who coordinates with the customer, particularly for ensuring that the customer understands critical technical detail and that a customer's desires are, in turn, communicated to the technical team. This is unchanged from Sheard's roles (1996).
Technical Manager	Individual who controls cost, schedule, and resources for the *technical* aspects of a system; often someone who works in coordination with an overall project or program manager. This is unchanged from Sheard's roles (1996).
Information Manager	Individual who is responsible for the flow of information during system development activities. This includes the systems management activities of configuration management, data management, or metrics. This is unchanged from Sheard's roles (1996).
Coordinator	Individual who brings together and brings to agreement a broad set of individuals or groups who help to resolve systems related issues. This is a critical aspect of the management of teams. This is unchanged from Sheard's roles (1996).
Instructor/Teacher	Individual who is provides or oversees critical instruction on the systems engineering discipline, practices, processes, etc. This can include the development or delivery of training curriculum as well as academic instruction of formal university courses related to systems engineering. While any discipline could conceivably have an instructor role, this denotes a focus on systems and is a critical component in the development of an effective systems engineering workforce. This is an addition to the Sheard roles (1996)

Systems Engineering Lifecycle

- **Concept Definition** - A set of core technical activities of SE in which the problem space and the needs of the stakeholders are closely examined. This consists of analysis of the problem space, business or mission analysis, and the definition of stakeholder needs for required services within it.
- **System Definition** - A set of core technical activities of SE, including the activities that are completed primarily in the front-end portion of the system design. This consists of the definition of system requirements, the design of one or more logical and physical architectures, and analysis and selection between possible solution options.
- **System Realization** - The activities required to build a system, integrate disparate system elements, and ensure that a system both meets the needs of stakeholders and aligns with the

requirements identified in the system definition stage. This includes integration, verification, and validation (IV&V).

- **System Deployment and Use** - A set of core technical activities of SE to ensure that the developed system is operationally acceptable and that the responsibility for the effective, efficient, and safe operations of the system is transferred to the owner. Considerations for deployment and use must be included throughout the system life cycle. Activities within this stage include deployment, operation, maintenance, and logistics.
- **Product and Service Life Management** - Deals with the overall life cycle planning and support of a system. The life of a product or service spans a considerably longer period of time than the time required to design and develop the system. This stage includes service life extension, updates, upgrades, and modernization, and disposal and retirement. The organizations in the current sample are primarily concentrated on new development, so this is a very under-represented aspect of the life cycle.
- In addition to these life cycle phases, the SEBoK includes orthogonal activities of systems engineers, **Systems Engineering Management**, defined as managing the resources and assets allocated to perform SE activities. Activities include planning, assessment and control, risk management, measurement, decision management, configuration management, information management, and quality management. These activities can occur at any point in the systems engineering lifecycle.

Career Path Self-Assessment Tool

Date: ___________________

Concept Definition										
System Definition										
System Realization										
System Deployment and Use										
Product and Service Life Management										
Systems Engineering Management										
Role(s) Performed										
Domain(s)										
System Characteristics										
Position										
Organization(s)										
Dates										
Milestones (Key positions, education, or training)										