SYSTEMS ENGINEERING
Research Center

Atlas: The Theory of Effective Systems Engineers, Version 0.25

Technical Report SERC-2014-TR-038-4

November 21, 2014

Research Team:

Dr. Art Pyster, Dr. Peter Dominick, Devanandham Henry, Nicole Hutchison, Carlo Lipizzi, Monis Kamil, Shipra Manchanda

Stevens Institute of Technology

Work conducted under Research Topics 45 and 106

This material is based upon work supported, in whole or in part, by the U.S. Department of Defense through the Systems Engineering Research Center (SERC) under Contracts H98230-08-D-0171 and HQ0034-13-D-0004. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology.

Table of Contents

List of Figures

List of Tables

ACKNOWLEDGEMENTS

The Helix team would like to thank the many organizations and individuals that have made this research possible through their active participation. Collectively, they contributed hundreds of hours to enable us to collect the information that is critical to the results reported here. We further thank the many members of the Helix Advisory Panel and the individuals who participated in the July 2014 Helix workshop. Their counsel on our research goals, strategy, and plans have been instrumental.

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 National Defense Industrial Association Systems Engineering Division (NDIA-SED) and the International Council on Systems Engineering (INCOSE) are both valued partners in this research and we thank them, especially Don Gelosh, Courtney Wright, David Long, and Bill Miller.

Finally, we will always be grateful for the time that the late Ricardo Pineda spent with the Helix team. His insights, energy, and the joy in how he approached life are missed.

EXECUTIVE SUMMARY

During this most recent phase of the Helix project, the team has developed *Atlas*[1]: the *Theory of Effective Systems Engineers.* Figure 1 provides an overview of the theory[2] and an outline of its key variables.

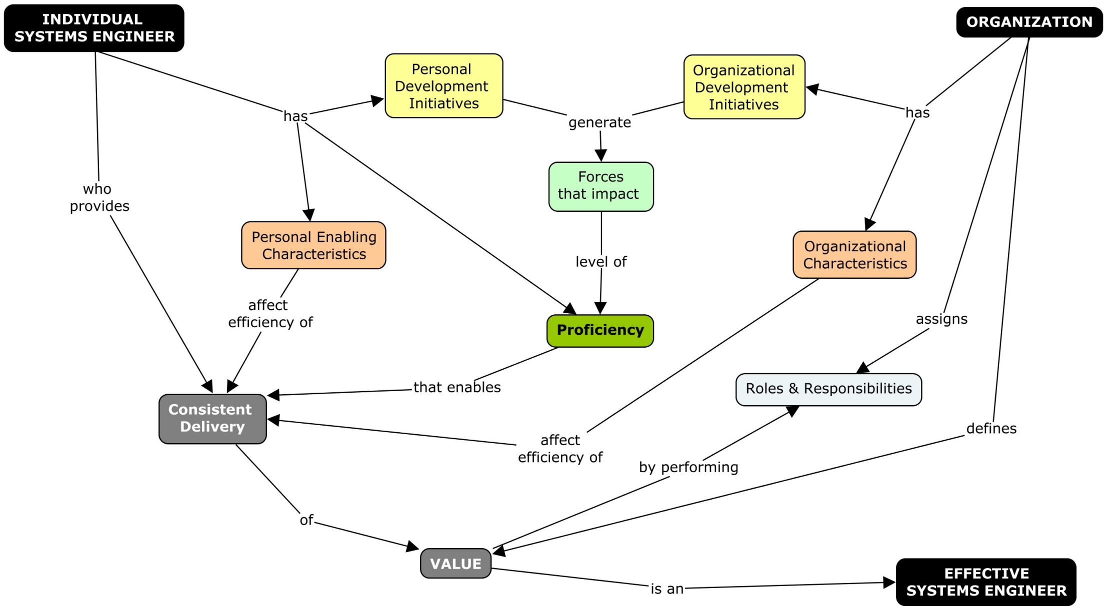


Figure 1. Overview of *Atlas*.

This report offers insight into proficiency, the forces that impact proficiency, and personal and organizational enabling characteristics. Proficiency is the quality or state of an individual's systems engineering relevant knowledge, skills, abilities, behaviors, and cognitions. The team has identified six proficiency areas that are further described in this report: **Math/Science/General Engineering**; **System's Domain and Operational Context**; **Systems Engineering Discipline**; **Systems Engineering Mindset**; **Interpersonal Skills**; and **Technical Leadership**. The forces that impact proficiency are **experiences**, **mentoring**, and **education and training**. The discussion of personal and organizational characteristics builds upon the findings detailed in the first and second Helix reports. (Pyster et al. 2013 and 2014)

Building on research reported in those earlier Helix reports, the Helix team performed additional career path analysis, including refinement of the criteria for defining junior, mid-level, and senior systems engineers. (See Section 7) These updated criteria help to provide more clarity on the growth of systems engineers, which is a critical step in beginning to model how systems engineers progress through their careers.

As its version number implies, *Atlas* v. 0.25 is an early draft of the theory. The Helix team will continue to refine it going forward.

---

[1] In Greek mythology, Atlas held the world on his shoulders and many of the systems engineers who were interviewed felt they held the project on their shoulders. Also, as a collection of maps or pathways or directions, the term "atlas" evokes the image of someone who guides the (technical) direction of a project.

[2] There are many definitions of "theory". Here it means a set of observations and principles that can be used to explain and predict a class of phenomena, in this case, the phenomena of systems engineers performing effectively, delivering value. (based on the definition by Miriam-Webster 2014)

1. INTRODUCTION

The US Department of Defense (DoD) and the Defense Industrial Base (DIB) – contractors that develop and deliver systems to the DoD – have been facing major systems engineering challenges in recent years (e.g. GAO 2008, 2011, 2012, 2013). Mission requirements are evolving and they demand ever more sophisticated and complex systems (e.g. Boehm et al. 2010; INCOSE Technical Operations 2007; Davidz and Nightingale 2007; Frank et al. 2007; INCOSE 2014); the tools, processes, and technologies that systems engineers must master keep changing more rapidly (e.g. Frank 2006); and budgets and schedules are being compressed dramatically. An additional concern is that thousands of systems engineers in the defense workforce are nearing retirement; they will take with them hundreds of thousands of staff-years of experience. (DoD 2013)

Organizations have responded to these challenges in a variety of ways, such as offering extended training and education to their current workforce or systematically seeking to select specialty engineers with the potential to become systems engineers and incorporating them into the ranks of systems engineers. Unknown is whether these actions are producing the desired results because there is no common understanding of the diverse roles that systems engineers play; how they are selected and evaluated; what competencies are most important for different roles; how to evaluate effectiveness; or how experiences impact effectiveness. These and many other insights will be critical to maintaining and growing the systems engineering workforce in the US DoD and DIB.

In order to provide these insights, the Systems Engineering Research Center (SERC), a US DoD University Affiliated Research Center (UARC), is conducting 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 Systems Engineering Division of the National Defense Industrial Association (NDIA-SED), and the International Council on Systems Engineering (INCOSE) 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 organizations.

Helix is a multi-year longitudinal research project, which is gathering data from many DoD and DIB organizations through a combination of techniques, including interviews with hundreds of systems engineers. In 2014, Helix has begun 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.

Helix research is culminating in a theory of effective systems engineers, which is being developed incrementally. This technical report, the third published by Helix, presents Version 0.25 of that theory, which the team calls *Atlas*.

There are many definitions of the term“theory”. Here it means a set of observations and principles that can be used to explain and predict a class of phenomena, in this case, the phenomena of systems engineers performing effectively, delivering value to the organizations for which they work. (based on the definition by Miriam-Webster 2014)

Building on previous work, *Atlas* is primarily based on extensive interviews conducted with 165 systems engineers from eleven organizations in 2013 and 2014, their resumes, and follow up interviews with 65 of these systems engineers. Future versions of *Atlas* will incorporate insights and validation from additional data collection and further analysis of data already collected.

Atlas identifies the key variables that impact a systems engineer’s effectiveness – positively or negatively – and provides details on how these variables impact effectiveness as much as possible. The insights provided in this report reflect the current state of the Helix data.

This report is organized as follows:

- Section 2 explains updates in the research methodology;
- Section 3 offers first thoughts on an emerging theory of systems engineers;
- Section 4 provides insight into the key proficiencies of systems engineers;
- Section 5 provides insight into the forces that impact systems engineers' proficiency and links proficiencies to the forces most commonly used to improve them;
- Section 6 provides details on personal and organizational enabling characteristics;
- Section 7 provides the current level of analysis on career paths for systems engineers; and
- Section 8 provides an overview of the future directions for Helix.

Throughout this report, the Helix team provides insights related to the career progression of systems engineers – namely how junior systems engineers over time grow into roles as senior systems engineers. The sample currently is heavy in senior systems engineers, with a smaller number of junior systems engineers, and very few mid-level systems engineers. Because of this, most insights reflect the junior and senior systems engineers in the sample. Going forward, the Helix team will continue to explore how systems engineers mature.

2. PHASE 4 METHODOLOGY

Although there have been small refinements, the Helix methodology has not changed significantly since the publication of the first Helix report in December 2013, as shown in Figure 1. In Phases 1 and 2 of the project, the team focused primarily on initial interviews and analysis with subsequent reporting (Steps A, B1-B3, C1-C3, D, and E). In Phases 3 and 4, the Helix team has focused data collection effort primarily around a few initial interviews (B3) and follow up interviews (B4), has conducted additional analysis (C4 and C5), and has reviewed the methodology (F), as highlighted in red in Figure 2. The team has also focused on the end stage goals for addressing the research questions (D) by beginning work on th theory of systems engineers, *Atlas* (D2). The methods used for each of these focus areas are outlined below.

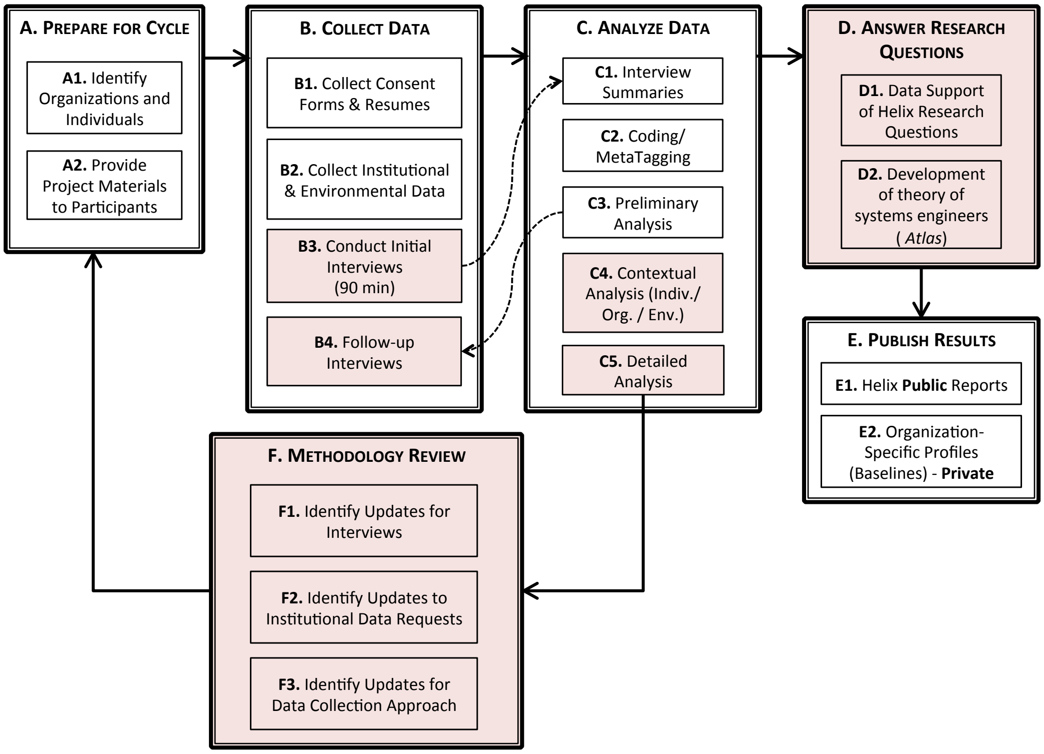


Figure 2. Helix research methodology overview.

2.1 INITIAL INTERVIEWS (B3)

In Phase 4, the Helix team conducted a small number of initial interviews with a new set of interviewees. Specifically, the team focused on consultants in systems engineering – individuals who were largely independent but have a variety of experiences across many domains, working with different types of problems and systems, and who have insight into different types of organizations. In particular, the Helix team used these interviews to validate findings in the data. The team identified patterns seen across organizations and focused on whether or not these patterns had also been seen by the consultants. In terms of the methodology, the only difference in these interviews is that they were kept to 60 minutes.

2.2 FOLLOW UP INTERVIEWS (B4)

Follow up interviews represent the majority of the new data added to Helix In Phases 3 and 4. The team piloted the follow up interview process, refined the questions, and has been working on full-scale follow up interviews. The follow up interview process is parallel to the initial interview process, with a few differences. At a high level, the interviews are shorter (limited to 60 minutes), are conducted over the phone instead of in person, and are all conducted individually (whereas many of the initial interviews are conducted in small groups). The steps for conducting follow up interviews include:

1. The Helix team creates a career profile (see Section 7 below). This is used to identify any questions about the interviewee's background, experiences, education, etc. that may not have been addressed in the first interview.
2. The team reviews the transcript or summary from the individual's first interview and identifies any points of confusion, concern, or areas of interest that need to be addressed in the follow up interview.
3. The team also asks questions of the individual over the phone. A standard set of follow up questions have been developed – tailored to the seniority of the systems engineer – that are asked during follow up interviews.

As with the initial interviews, data collection is done in one of two ways: when recording is allowed, a transcript is created and data analysis is done using this transcript; when recording is not allowed, the team takes detailed notes during these interviews, cleans up the notes, and uses these for data analysis.

2.3 ADDITIONAL ANALYSIS (C4 AND C5)

Beginning in Phase 3, the Helix team added a new aspect to its analysis: the use of demographic, background, and interview data to develop career profiles for the individuals who have participated in the study. The team used this mix of information to develop a profile that explains how each individual's career has changed over time. When possible, the team used interview data (from either initial or follow up interviews) to determine the reasons for some of these changes. This work continued into Phase 4 and the team has nearly doubled the number of individuals included in this analysis.

There are several aspects being examined in the career path analysis, including current seniority levels of interviewees and multiple aspects of each individual's experiences such as the stages of the lifecycle, types of programs, types of systems, and areas of responsibility an individual has been exposed to and the roles he or she has played.

Additional details on career path analysis are found in Section 7.

2.4 METHODOLOGY REVIEW (F)

The team has reviewed its records of the methods used during interviews and for analysis to date. The team believes that these are still sound, but has made a few minor updates to the specific questions or tools used for data collection in order to achieve more consistent or meaningful results. Specifically, the team revised the questions for the initial interviews by:

- Reviewing the way questions were asked in interviews over time and, in the event of multiple methods for asking a question (i.e. different phrasing), determining which of the options provided the best or most reliable answers;
- Identifying questions which have not traditionally provided clear or useful answers and either modifying these or substituting additional questions, which were piloted in the follow up interviews; and

- Identifying new questions from the follow up interviews that should become part of the initial interview.

Going forward, the team will further update the interview questions to support *Atlas* validation.

2.5 MOVE FROM GROUNDED THEORY TO A MORE INTENTIONAL RESEARCH FOCUS

The Helix team began with a modified grounded theory approach. Grounded theory requires entering the research without any pre-conceptions about possible results. The team reviewed all data with as unbiased a perspective as possible, focusing on looking for patterns and anomalies to guide further analysis. After nine months of data collection and review, the team identified patterns in the data and particular areas of interest for the project. Some of these patterns included:

- The different paths taken by individuals in order to become and mature as systems engineers;
- The role of experiences in developing systems engineers;
- The role of mentoring in the maturation of systems engineers;
- The differences and similarities between systems engineering and those who perform systems engineering in government and industry;
- The use of processes to supplement experience and the balance between process and critical thinking; and
- The definition of effectiveness for systems engineers, including critical behaviors, and how that can be assessed outside of standard human resource style processes.

During Phase 4, the team began focusing on these areas of interest. Early examples of this are the analyses of mentoring and career paths, found in Sections 5 and 7, respectively. The Helix team will continue to investigate these areas because they are crucial components of *Atlas*.

2.6 ANSWER RESEARCH QUESTIONS (D1 AND D2)

In its first two reports, the Helix team linked the preliminary findings to the research questions, and this work continued in Phase 4. This report offers a preliminary version of a theory that would help to explain systems engineers as a whole, and would provide actionable information and data-based guidance for workforce development in systems engineering.

The theory is designed to provide insight on which variables most impact a systems engineer’s effectiveness (see Figure 2 in Section 3.2 below) and to provide detail on the specific knowledge, skills, abilities, behaviors, and cognitions that are critical to systems engineers (see Section 4).

The Helix team began development *of Atlas* in Summer 2014 by reviewing the data collected to date and identifying factors discussed by interviewees as either enablers or detractors to a systems engineer’s effectiveness. The team then began to identify the ways in which these variables interacted to produce a net impact on effectiveness.

On July 23, 2014, the Helix team hosted the first Helix workshop. Representatives from organizations that have participated in Helix, officers from the International Council on Systems Engineering (INCOSE), members of the Helix Advisory Panel, and other experts in the field attended the workshop. During this workshop, the participants generally agreed that the draft theory was reasonable and seemed to take into account the key variables that would impact a systems engineer’s effectiveness.

The Helix team took the results of the workshop and began to explore areas of interest or areas where there was less information when conducting additional interviews in summer and fall 2014. The team reviewed all Helix data to further refine the draft theory and to fill in as many details as possible.

Going forward, the Helix team will continue to refine *Atlas* in several ways:

- Follow up interviews will include questions that probe parts of the theory not discussed in initial interviews;
- Some initial interviews will be focused exclusively around the theory;
- The Helix team will work with individual systems engineers playing specific roles to determine what knowledge, skills, abilities, behaviors, and cognitions are most relevant for these roles, creating common profiles for typical systems engineering roles;
- The Helix team will coordinate with a few organizations to work through how implementation of the theory would work; and
- The Helix team will hold a second workshop focused around further refinement of the theory.

This will form the basis of version 0.5 of *Atlas*, which will be reported in late 2015.

3. *ATLAS*: A THEORY OF EFFECTIVE SYSTEMS ENGINEERS

According to Merriam-Webster, a theory is a set of general principles or ideas relating to a particular subject. (2014) In the case of the Helix Project, the theory being developed is one that is intended to explain what makes systems engineers effective and why and provide insight into how individuals can develop into effective systems engineers throughout their careers. The first section in this chapter provides an overview of *Atlas*, with subsequent sections providing detail on particular aspects of the theory.

Throughout the discussion of *Atlas*, the terms **junior**, **mid-level**, and **senior** systems engineer will be used. In general, junior systems engineers are those new to the field and fairly early in their careers, while senior systems engineers have been in the field at least 15 years and tend to provide guidance on high-level systems engineering issues. The distinctions between these groups – and the ways in which junior systems engineers may grow into the roles of senior systems engineers – are explored in the text wherever possible.

3.1 DEVELOPING A THEORY OF EFFECTIVE SYSTEMS ENGINEERS

The theory is being developed in three incremental releases:

- **Version 0.25**, presented in this report, is the first published draft. Based on data collected to date, it proposes the key variables that explain effectiveness and offers a preliminary explanation of how those variables impact effectiveness.

- **Version 0.5** will offer more detail in the description of the variables, particularly in how these variables are related and how they may be assessed. The theory will be mature enough that an individual or an organization could implement it with some teaching and coaching from the Helix team. It will also include some validation using data from outside the DoD. The team anticipates that it will be published in late 2015.

- **Version 1.0** will be a mature version of the theory, packaged in a way that is independently implementable. It will include guidance on how to assess the collective proficiency of an organization's workforce, offer additional validation with longitudinal data. The team anticipates that this will be published in late 2016.

Some aspects of *Atlas* that will mature over time include ways in which the theory can be applied to group settings – from small teams to the overall systems engineering workforce. In addition, expected changes for an individual over time will be explored. For example, many senior systems engineers stated that they were able to do complex physics-based analysis early in their careers and that they still understand the results of these analyses, but that it would take a substantial amount of time and energy for them to be able to perform those analyses today. This indicates that required proficiency levels change not only with roles, but also for an individual over time. The Helix team will explore these issues more deeply in subsequent data collection and analysis.

3.2 *ATLAS* OVERVIEW

Figure 3 is a systemigram offering a high-level view *of Atlas.* The two primary actors in this model are the individual systems engineer and the organization. The primary focus of the view is understood by reading the path from upper left hand corner to lower right hand corner – **an individual systems**

engineer who provides consistent delivery of value within the context of his or her organization is an effective systems engineer.

"Value" is context-dependent. There are likely some values provided by systems engineers regardless of the organizational context. Data from the Helix interviews to date indicates that some such values may include:

- Correctly identifying the needs of the customer;
- Providing clarity on the system vision;
- Communicating the system vision; and
- Delivering on the needs of the customer.

In addition to generic values, each organization will define what it most values in systems engineers.

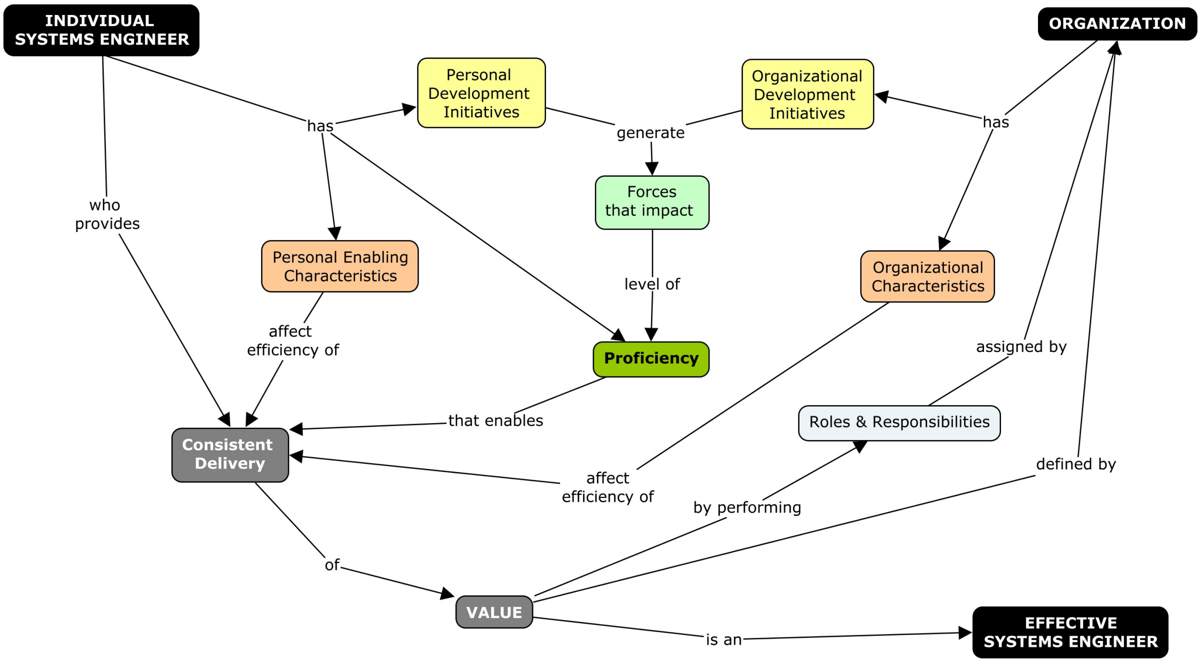


Figure 3. Top-level view of *Atlas*

Proficiency is the quality or state of knowledge, skills, abilities, behaviors, and cognitions. Each individual systems engineer has a level of proficiency in the knowledge, skills, abilities, behaviors, and cognitions that enable being effective as a systems engineer and delivering value. Though which values are relevant may differ between organizations and based on specific roles, the Helix team has used the data from the interviews to identify proficiencies that enable delivery of all major values identified to date. These proficiencies are described in detail in Section 4. In an ideal situation, an individual with the right proficiencies will by definition provide consistent value and, therefore, be effective. However, there are other factors that enable or inhibit an individual's effectiveness.

Each organization not only defines which values are most relevant, but also assigns roles and responsibilities to its systems engineers. If an individual's role or responsibilities do not align with his or her proficiencies, he or she may have difficulty delivering value. It is important to note that having the role of a systems engineer does not necessarily mean that an individual will have the *title* of systems engineer. In fact, only 21% of the systems engineers in the Helix sample currently have the title "systems engineer" and 47% of the interviewees have *never* held that title formally. For additional detail, please see the career path analysis in Section 7.

There are also forces that will impact a systems engineer's effectiveness. Based on the Helix data to date, the three primary forces that impact effectiveness are *experiences*, *mentoring*, and *education and training*. Each force is discussed in more detail in Section 5. These forces may come about organically, but there are also initiatives that are intended to improve the proficiency of a systems engineer. These may be either formal initiatives from the organization or personal initiatives that a systems engineer may pursue to improve his or her abilities. For example, an organization that believes diversity of experiences is important for the development of a systems engineer may decide to create a rotational program for its systems engineers or an individual may decide to pursue a master's degree in systems engineering to improve his or her knowledge.

Finally, there are certain characteristics of both the individual and the organization that have an impact on the consistent delivery of value that have to be taken into account. Personal enabling characteristics include things such as learning ability, ethics and morality, professionalism, curiosity, motivation, self-awareness, and diligence. Organizational characteristics include aspects such as culture, structure, values, appreciation of systems engineering, definition of systems engineering and systems engineer, rewards and recognitions systems, and career growth potential. The Helix team has data in the current sample on the characteristics – particularly as it pertains to organizational culture, values, and relationship with SE – but in general this is an underdeveloped area of the study. Going forward in 2015, the Helix team will improve data collection and analysis in this area.

Atlas is the overarching theory of the effectiveness of systems engineers. The next three sections will provide additional detail on different aspects of that theory.

4. PROFICIENCY OF SYSTEMS ENGINEERS

In every interview conducted by the Helix project, participants discussed critical knowledge, skills, abilities, and patterns of thinking or abilities to learn – cognitions – that are critical to the development of systems engineers. These items are the "proficiencies" discussed in Figure 3, above. In order to help systems engineers and the organizations that employ them understand proficiencies, the Helix team has developed a model of proficiencies as part of *Atlas*, which is generically outlined in Figure 4, below.

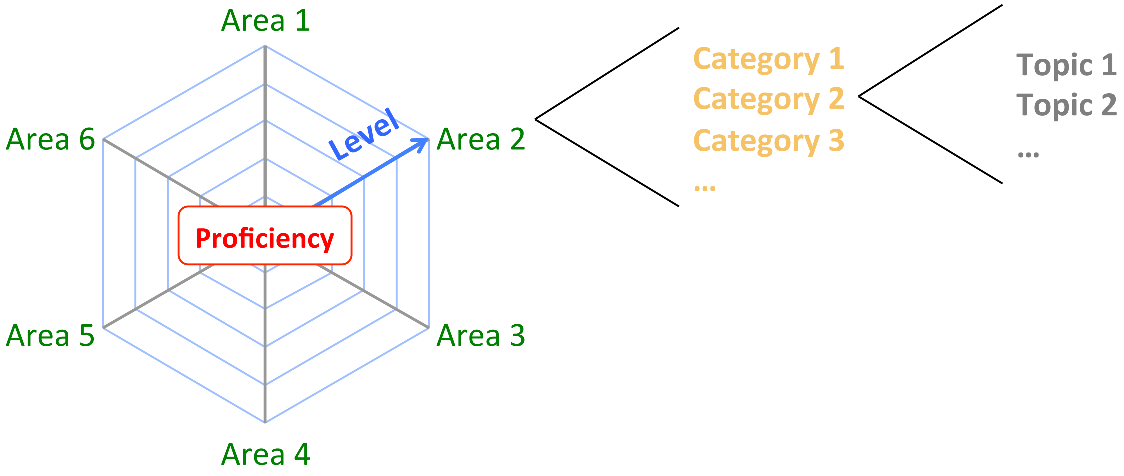


Figure 4. Generic proficiency framework.

As shown in Figure 4, there are several aspects to the generic proficiency framework:

- **Proficiency** is the quality or state of knowledge, skills, abilities, behaviors, and cognitions.
- **Proficiency areas** are groupings of related knowledge, skills, abilities, behaviors, and/or cognitions.
- Each Proficiency Area includes between four to six **categories**, which are specific types of knowledge, skills, abilities, behaviors, and cognitions with shared characteristics.
- Some categories are further refined into **topics**, which are the most discrete areas of knowledge included in the model. Topics where developed for areas where interviewees focused a great deal of discussion and detail.
- For each proficiency area, there are multiple **proficiency 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 cognitions.

There are many different types of proficiencies that are useful for systems engineers. The Helix team has created six proficiency areas, which are discussed in the remainder of this section. For each proficiency area, categories are outlined and, where appropriate, topics are also discussed. Currently, not all categories contain topics. In general, topics were created where the existing Helix data provided considerable detail or depth of discussion within a category. It is possible that even in subsequent versions of *Atlas* not all categories will have explicit topics.

For this version of *Atlas*, the proficiency levels are presented at the category level. Going forward, the

Helix team will determine specific criteria to define proficiency for each topic as well as how topics and categories roll up into an overall proficiency level for each proficiency area.

Using this framework, individual systems engineers or systems engineering organizations can assess current levels of systems engineering proficiencies. In addition, they can compare these profiles to the typical or desired proficiency for a given role. This will allow any gaps between current and required proficiency to be identified. The Helix team believes this will provide an important tool for creating career plans for the development of systems engineers.

4.1 MODELING THE PROFICIENCY OF SYSTEMS ENGINEERS

The generic proficiency framework was applied to create the proficiency model of *Atlas,* found in Figure 5, below. Each area is discussed in more detail below.

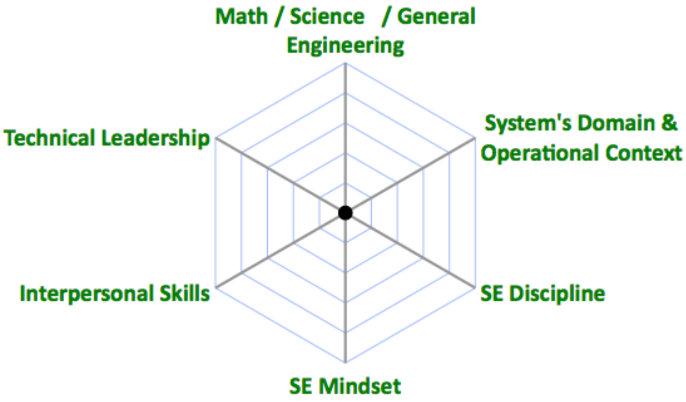


Figure 5. Proficiency Areas for Systems Engineers

These proficiency areas were identified based on the data collected through Helix interviews:

- **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.

The proficiency areas are a mix of the more traditional technical content related to engineering, systems engineering-specific content, and non-technical skills. This mix is the reason that the Helix team has chosen to use the term "proficiency" instead of "competency", even though some areas, categories, and topics may overlap with existing SE or related competency models. Where possible, the Helix team has utilized models and frameworks from existing literature to help further refine each proficiency area into categories and, where appropriate, topics. This is an evolution of the Helix methodology. The data collected and analyzed using grounded theory methods has been compared to the existing literature to build the *Atlas* proficiency model.

4.2 AREA: MATH/SCIENCE/GENERAL ENGINEERING

When asked what skills were critical for developing into a successful systems engineer, most interviewees stated that understanding math, science, and general engineering is a critical foundation. When asked about the details – which specific aspects of math, science, and general engineering are most critical – few individuals could provide specific recommendations beyond what was needed for the specific systems they were developing or their particular roles. For example, in an organization that focuses on aerospace systems, physics was mentioned as particularly important and individuals who perform analysis discussed the importance of statistical analysis.

Because the data does not yet provide more general principles, the Helix team has reviewed the *Graduate Reference Curriculum for Systems Engineering (GRCSE®)*, which defines the types of prerequisite knowledge individuals should have before entering a master's program in systems engineering. (Pyster and Olwell et al. 2012) GRCSE is intended to define the educational requirements that will provide the foundation for a systems engineering student and, eventually, for a practitioner. Because GRCSE was produced as a collaborative and community-based resource, the Helix team believes it serves as a suitable starting point for the specific categories that are important for systems engineers.

The proficiency area Math/Science/General Engineering contains foundational concepts from mathematics, physical sciences, and general engineering. The specific definitions for these categories, from GRCSE, are:

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

- **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).

- **Natural Science Foundations.** Basic concepts and principles of one of the natural science disciplines (e.g., physics, biology, chemistry, etc.). This will provide students with a technical background in at least one scientific domain area. This preparation should include laboratory work that involves experimental techniques, the application of the scientific method, and comprehension of appropriate methods for data quality assurance and analysis.

- **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. Students should have some level of practical experience, whether through capstones, internships, or course projects. Practical experience should include the application of engineering fundamentals in a specific domain context.

- **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).

The Helix team uses Bloom's taxonomy to define the proficiency levels for Math/Science/General Engineering, as described below.

4.2.1 BLOOM'S TAXONOMY

Math/Science/General Engineering is defined in terms of Bloom's levels of attainment (Bloom et al. 1956). Bloom's levels define an objective level of achievement. Though defined in an academic setting, this can be useful for defining the skill level required in a professional environment as well. The different levels of Bloom's taxonomy *Atlas* uses are defined in Table 1, below.

Table 1. Levels in Bloom's Taxonomy (from Pyster and Olwell at al. 2012)

Level	Competency
Knowledge	**Ability to remember previously learned material**. Test observation and recall of information; i.e., "bring to mind the appropriate information" (e.g., dates, events, places, knowledge of major ideas, and mastery of subject matter).
Comprehension	**Ability to understand information and ability to grasp meaning of material presented** (e.g., translate knowledge into new context, interpret facts, compare, contrast, order, group, infer causes, predict consequences, etc).
Application	**Ability to use learned material in new and concrete situations** (e.g., use information, methods, concepts, and theories to solve problems requiring the skills or knowledge presented).
Analysis	**Ability to decompose learned material into constituent parts in order to understand structure of the whole**. This includes seeing patterns, organization of parts, recognition of hidden meanings, and the identification of parts.
Synthesis	**Ability to put parts together to form a new whole**. This involves the building upon existing ideas to create new ones, generalizing from facts, relating knowledge from several areas, and predicting and drawing conclusions. It may also involve the adaptation of "general" solution principles to the embodiment of a specific problem.
Evaluation	**Ability to pass judgment on value of material within a given context or purpose**. This involves making comparisons and discriminating between ideas, assessing the value of theories, making choices based on reasoned arguments, verifying the value of evidence, and recognizing subjectivity.

In version 0.25 of *Atlas*, *Math/Science/General Engineering* clusters the six Bloom's levels into three broad category levels as shown in Table 2, below. For example, a proficiency of "knowledge" in natural science foundations would indicate that an individual has the ability to recite elements such as the scientific method, the basic laws of physics, etc. If an individual could then use these things in a real world context and allow them to inform his or her work, then the individual would have an "Application" level of proficiency.

Table 2. ***Math/Science/General Engineering Categories &*** **Proficiency Levels**

MATH/SCIENCE / GEN. ENG.	Low	Medium	High
Probability and Statistics	Knowledge, Comprehension	Application, Analysis	Synthesis, Evaluation
Calculus & Analytical Geometry	Knowledge, Comprehension	Application, Analysis	Synthesis, Evaluation
Natural Science Foundations	Knowledge, Comprehension	Application, Analysis	Synthesis, Evaluation
Engineering Fundamentals	Knowledge, Comprehension	Application, Analysis	Synthesis, Evaluation
Computing Fundamentals	Knowledge, Comprehension	Application, Analysis	Synthesis, Evaluation

Although almost every interviewee stated that foundations in math, science, and engineering were important, they provided few details. The few specifics provided tended to be related to the systems developed by the organization. For example, at an aerospace-focused organization, a solid understanding of physics was frequently mentioned. However, individuals at an electronics-focused organization did not mention physics in their interviews. As work continues, the Helix team will more explicitly explore the aspects of *Math/Science/General Engineering* required at different stages in systems engineers' careers.

Senior systems engineers frequently stated that having a high level of attainment in *Math/Science/General Engineering* was more critical early in their careers than later in their careers. For example, early in their careers those interviewed performed mathematical calculations and used physics principles on a regular basis in their jobs. Now, younger engineers perform those actions instead and senior systems engineers tend oversee them and review and guide their work. Today, senior engineers need to understand these things well enough to have an instinct about whether these things are done correctly and whether the outputs make sense, but, they no longer perform the detailed steps.

The Helix team will explore more specifically the skills associated with these foundational categories going forward and will also explore how the required proficiency levels change based on role and progression through a systems engineers' career.

4.3 AREA: SYSTEM'S DOMAIN & OPERATIONAL CONTEXT

The second proficiency area is *System's Domain & Operational Context*, which contains the relevant domains, disciplines, and technologies for a given system and its operation. This proficiency area is heavily tied to the system(s) on which an individual is working. If that individual transitions to a new system, his or her proficiency level may change depending on familiarity with the new relevant domains, technologies, and disciplines. The categories for this proficiency area are defined below:

- **Relevant Domains** – The higher-order domain for the *current* system. In this context, *domain* means the overarching sphere of technology. This includes things such as space, aerospace, marine, communication, finance, etc., and is a measure of how proficient the individual is within the domain. Proficiency in 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 communication systems.

- **Relevant Technologies & Systems** – Within the context of the system's domain, there are specific technologies or enabling or support systems that are particularly relevant. For example, on a marine system, there may be gas turbine, radar, and sonar systems, and each technology has it's own terminology, challenges, etc. This category reflects how familiar an individual is with the relevant technologies on a system.
- **Relevant Disciplines** – Skill in the relevant related disciplines for a particular system. For example, for a communications system, electrical engineering will be an important discipline to understand while civil engineering will be less relevant. It is likely that there may be more than one relevant discipline.
- **Familiarity with System's Concept of Operations (ConOps)** –The individual's understanding of the system's concept of operations.

Almost all interviewees stated that these categories are critical for a systems engineer to be effective. General statements such as it is "important to understand how the system will be used because it helps you make better design decisions," were common. And in a similar vein to *Math/Science/General Engineering*, *System's Domain & Operational Context* is impacted by the business of the specific organization.

Based on the data to date, the Helix team has developed a description of what it means for someone to have "high" proficiency in *Systems Domain and Operational Context* categories, as described in Table 3, below. In general, low proficiency in this area was discussed only in terms of lacking the skills described as "high" proficiency. Descriptions of the "medium" proficiency level are notional and based on the team's experiences and inferences from the data. The Helix team will therefore focus additional data collection on defining the different proficiency levels in this area.

Table 3. ***Systems Domain and Operational Context*** **Categories & Proficiency Levels**

SYSTEM'S DOMAIN & OPERATIONAL CONTEXT	Low	Medium	High
Relevant Domains	• Has a solid grasp of general engineering terminology, but has limited familiarity with the terminology of the domain • Has a cursory understanding of the critical engineering disciplines for the system's domain • Has demonstrated ability in one of the critical disciplines for the domain	• Is familiar with the more common terminology that is unique to the system's domain, but will require guidance on more obscure or specialized areas • Has a limited understanding of the critical engineering disciplines for the system's domain • Has depth of experience in at least one critical	• Has mastery of the terminology that is unique or particularly relevant to the system's domain • Has a solid understanding of the critical engineering disciplines for the system's domain • Has expertise in at least one critical engineering discipline for the system's

SYSTEM'S DOMAIN & OPERATIONAL CONTEXT	Low	Medium	High
	• Has limited familiarity with the key technical considerations, pitfalls, pitfalls, etc. for the system's domain	engineering discipline for the system's domain • Is somewhat familiar with the key technical considerations, pitfalls, pitfalls, etc. for the system's domain	domain • Has a mastery of the key technical considerations, pitfalls, pitfalls, etc. for the system's domain
Relevant Technologies & Systems	• Has cursory understanding of at least one technology or system that is particularly relevant to the domain	• Conversational understanding of the technologies or systems that are particularly relevant • Has depth of experience in at least one technology or system that is particularly relevant to the domain	• Somewhat detailed understanding of the technologies or systems that are particularly relevant • Expertise in at least one of the technologies or systems that are particularly relevant
Relevant Disciplines	• Has a basic understanding in at least one critical discipline for the system; e.g. electrical, mechanical, software, etc.	• Has deep experience in at least one critical discipline for the system; e.g. electrical, mechanical, software, etc. • Has a basic understanding methods, tools, and terminology for each of the critical disciplines for the domain	• Has deep experience in at least one critical discipline for the system; e.g. electrical, mechanical, software, etc. • Has a conversational understanding of methods, tools, and terminology for each of the critical disciplines for the domain
Familiarity with the System's CONOPS	• Has general awareness of the concept of operations	• Participated in developing the concept of operations	• Led the development of the concept of operations

The specific discussion of how proficiency in each of these categories "rolls up" into an overall proficiency level for the area will be determined in the next release of *Atlas*. If, for example, an individual is considered "high" in all categories, it would be easy to assign that person as "high" in *System's Domain and Operational Context*. If, however, an individual were a "medium" in some areas and "high" in others, it is more difficult to place his or her overall proficiency.

There are additional factors that may be incorporated into this proficiency area going forward. For example, in follow up interviews, many interviewees who worked in multiple domains indicated that experiences in a variety of domains is valuable because it enables one to develop patterns of best practices and potential challenges that are somewhat independent of the domain. This is different from depth in a domain, which is discussed above. The exact nature of how this will be incorporated and balanced with depth in relevant domains will be determined in *Atlas v. 0.5*.

According to the Helix interview data, individuals can improve proficiencies in this area primarily through experiences and mentoring, as discussed in the subsequent "forces" sections, below.

The specific domains, disciplines, and technologies that are related to the current system will vary based on context, but an individual's familiarity with these elements is an important part of his or her proficiency and should be taken into account by anyone hoping to understand his or her potential.

4.4 AREA: SYSTEMS ENGINEERING DISCIPLINE

The third proficiency area is *Systems Engineering Discipline*. Because this area includes a wide variety of information, three of the categories, highlighted with an * in Table 4, are further split into topics. The categories below were developed based on data from Helix interviews about critical systems engineering knowledge and skills. The names of the categories come from the *Guide to the Systems Engineering Body of Knowledge (SEBoK), version 1.3*. (BKCASE Authors 2014)

- **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. For further detail, see Section 4.4.1.
- **SE 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 2014) For further detail, see Section 4.4.4.
- **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) This includes understanding and ability to apply systems engineering processes, tailoring it as appropriate to match the goals, constraints, and character of a given system. For further detail, see Section 4.4.2. Processes generally refer to the specific guidelines an organization develops for implementing systems engineering methods and tools refer to software programs that are designed to support systems engineering activities.
- **System Complexity** – *The degree to which a system's design or code is difficult to understand because of numerous components or relationships among components.* (ISO/IEC 2009) In this context, the proficiency is the level of complexity an individual can handle. This may include things such as their ability to work on components, subsystem, systems, or systems of systems as well as the ability to work on products, platforms, services, or enterprises. Note that others have also equated a system's complexity with the degree to which that system's behavior cannot be predicted. There are numerous measures of systems complexity in the literature and at this point in time, the Helix team does not have a generic measure of complexity.

Table 4. ***Systems Engineering Discipline*** **Categories & Proficiency Levels**

SYSTEMS ENGINEERING DISCIPLINE	Low	Medium	High
Lifecycle*	• Has worked in at least two aspects of the system lifecycle • Is well-versed in at least one aspect of the system lifecycle • Has experience in systems under only one lifecycle model	• Has experience in at least 3 aspects of the lifecycle • Can lead at least one aspect of the lifecycle • Has familiarity with at least two different lifecycle models and when each is generally more appropriate	• Has experience in at least 4 aspects of the lifecycle • Can lead at least two aspects of the lifecycle • Can determine which lifecycle models are most appropriate for a given system • Can tailor lifecycle models to fit the business and operational context
SE Management	• Understands the appropriate approach for at least one SE management activity (e.g. planning, risk management, configuration management, assessment and control, or quality management) • Understands the importance of SE Management activities • Understands the basic terminology related to these activities	• Has deep experience in at least one SE management activity; e.g. planning, risk management, configuration management, assessment and control, or quality management • Understands the appropriate approaches for the other SE management activities • Understands the importance of these activities at different stages throughout the system's lifecycle	• Has deep experience in at least one SE management activity; e.g. planning, risk management, configuration management, assessment and control, or quality management; and can lead a team in this activity • Can communicate to a team the importance of these activities at different stages throughout the system's lifecycle

SYSTEMS ENGINEERING DISCIPLINE	Low	Medium	High
SE Methods, Processes, and Tools	• Generally focuses on isolated problems, not on system-level issues • Has a general awareness of modeling and simulation terminology and approaches • Has limited familiarity with organizational processes • Has enough familiarity with the SE tools required to perform tactical-level tasks with supervision	• Can help negotiate technical solutions between system elements • Can build complex system models and create simulations that provide valuable insight into system behaviors; has limited familiarity with organizational processes • Can navigate relevant processes within his or her organization with guidance • Can effectively oversee the utilization of systems engineering tools	• Can balance understanding the "best" technical solution for a system element with the "best" technical solution for the system overall • Understands the validity of system models and the results of simulations and can provide guidance on ways to improve these • Can successfully navigate all processes within his or her organization • Understands the uses, inputs, and outputs of systems engineering tools and is able to ensure that the workforce has access to them
System Complexity	• Can work on fairly linear problems • Can work on smaller scope elements, such as system components	• Can work on larger scope elements, such as subsystems • Can anticipate potential interface challenges	• Can manage systems with high numbers of interfaces • Can anticipate and respond effectively to emergent behavior

4.4.1 CATEGORY: LIFECYCLE

The intent of the *Lifecycle* category is to identify an individual's ability to work within and across different aspects of a system's development, use, and disposal. The topics contained in *Lifecycle* are defined below. Again, these terms come from the SEBoK (BKCASE Editorial Board 2014), but where the SEBoK quotes another source, that source is also included:

- **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)

  As a proficiency category, lifecycle models defines a systems engineer's familiarity with and understanding of multiple models, such as 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). The highest level of proficiency for this category is the ability to select an appropriate lifecycle model for a given system and to tailor the general lifecycle model to meet the environmental, organizational, and operational context of the system.

Interviewees stated that exposure to multiple models is important because it generates perspective and enables a systems engineer to build the skills and insights required to develop tailoring abilities. In general, senior systems engineers expressed concern about junior systems engineers being exposed to only one type of lifecycle, stating that this caused them to simply accept the lifecycle models and did not prompt them to ask questions about why the models were structured in a certain way or when different structures would be appropriate. Senior systems engineers also stated that one of the values they provide is guidance on which lifecycle model is appropriate for a given system.

- **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 2014) This consists of analysis of the problem space, business or mission analysis, and the definition of stakeholder needs for required services.

  Most senior systems engineers who were asked to discuss *Concept Definition* stated that it was more appropriate to mid-level and senior systems engineers because it generally involves dealing directly with stakeholders and understanding business processes. The senior systems engineers felt that, in general, junior systems engineers did not have the skillset required to do this effectively, though they could learn by supporting senior personnel in this area. Interestingly though, based on the Helix career path analysis, roughly the same percentage of the junior and senior systems engineers in the sample have worked in *Concept Definition* (40% and 38%, respectively). (See Section 7 for additional detail.) This is primarily due to rotational programs for junior personnel, which as stated above, some senior systems engineers feel might not provide sufficient experience.

- **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 2014) 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.

  About 60% of the systems engineers in the sample had their first role in systems definition. This is not surprising as almost all of the senior and mid-level systems engineers in the sample, and some of the junior systems engineers, started their careers doing specialty engineering. Many of them initially focused on detailed design work, which is generally agreed to be a very common first role for engineers who eventually work in systems. Senior systems engineers in particular stated that detailed design work is critical to growing an engineer's technical depth and that this depth then provides the technical expertise for practicing systems engineers. This phase also includes developing system requirements (from stakeholder requirements). In general, senior systems engineers oversee system requirements generation, but junior systems engineers are involved in the mechanics of requirements database management. Around 93% of the individuals in the Helix sample have performed systems definition activities at some point during their careers. (See Section 7 for additional detail.)

- **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 2014) This includes implementation as well as integration, verification, and validation (IV&V).

  Many of the systems engineers in the sample have worked in testing, which is a common point of entry into *System Realization*. As detailed design work provides technical depth as a basis for systems definition, testing provides detailed technical knowledge that is relevant for proficiency in System Realization. Almost 90% of the sample has worked in *System Realization* at some point. (See Section 7 for additional detail.)

  Senior systems engineers generally stated that proficiency in *System Realization* led to a higher level of proficiency in *System Definition*, because individuals could better understand how decisions made in *Systems Definition* would impact activities in *System Realization*.

- **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 2014) Considerations for deployment and use must be included throughout the system lifecycle. Activities within this stage include deployment, operation, maintenance, and logistics.

  Many systems engineers in the sample with proficiency in *System Deployment and Use* – 55% in the career analysis sample – gained this through operating fielded systems. The majority of these individuals were in the military and so experienced these systems directly in their operating environment, while a smaller number performed maintenance on systems in operation either as members of the military or as civilian DoD employees.

  Those with *System Deployment and Use* experiences were asked in follow up interviews why this proficiency is valuable. They explained that having hands-on experiences with systems in operation gave them significant insights into what the real needs and concerns of the end users were and that when they later worked on system design and test, this allowed them to create design considerations based on real world issues experienced in the field. This also helps proficiency in *System Realization*, as failure experiences in the field helped the systems engineers have a better idea of which types of testing will be most critical in verification and what areas of stakeholder concern will most likely come up in validation.

- **Product and Service Life Management** – *Deals with the overall lifecycle planning and support of a system*. (BKCASE Authors 2014) The life of a product or service often 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; based on current information, this seems to be a very under-represented aspect of the lifecycle.

Overall proficiency in the *Lifecycle* category is expected to be a combination of an individual's ability to work within different areas of the system lifecycle as well as to tailor lifecycle models and processes to appropriately for a given system. For example, because even junior systems engineers have typically had experience in at least two aspects of the systems lifecycle, to even have a "low" proficiency in this category the individual may need to have performed in at least one area. The specific way to determine an overall proficiency for this category will be developed for *Atlas v. 0.5*.

From an overall workforce perspective, this is an area where it is not necessary or sometimes even desirable for one individual to have expertise in all of these areas. For example, even chief systems engineers in the sample had high proficiency in some areas such as system definition and system realization, but may have only medium proficiency in other aspects of the lifecycle. Many interviewees working on larger systems – e.g. US DoD ACAT I – indicated that this is an area where it is critical for the overall team to have expertise in these areas, but that this expertise could come from more than one source. For small teams, where there is only systems engineer, then the level of proficiency may need to be more consistent across all topics in this category. Understanding the implications of applying this model to teams as well as individuals will be an important aspect of developing *Atlas v. 0.5.*

The primary mechanism for improving proficiency in this area is through experiences, though mentoring may be important, specifically in understanding the aspects of tailoring. Education and training can provide a basic understanding for these elements, including generic processes and terminology. For additional detail, please see the discussion on *forces* in Section 5.

4.4.2 CATEGORY: SYSTEMS ENGINEERING MANAGEMENT

The topics contained in the *Systems Engineering Management* category are defined below. The definitions for these terms come from the SEBoK, so that consistent terminology can be applied across the data set; where the SEBoK definition cites other sources, these citations are also included:

- **Planning** – *is performed concurrently and collaboratively with project planning. It involves developing and integrating technical plans to achieve the technical project objectives within the resource constraints and risk thresholds. Planning 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 2014)

- **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)

- **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)

- **Assessment and Control** – *The SEAC 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 2014)

- **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 2014)

Each aspect of *SE Management* was mentioned in Helix interviews, although all were mentioned at a generally superficial level, with no specific discussion of what it means to be proficient or the benefits proficiency in each of these topics. The Helix team will explore these topics in more detail going forward.

Again, Helix data indicates that proficiency improvements in these topics come primarily from experiences, although mentorship and education and training may also be helpful. See Section 5 for additional information.

4.4.3 CATEGORY: SYSTEMS ENGINEERING METHODS, PROCESSES, AND TOOLS

The topics contained in the *SE Methods, Processes, and Tools* category are outlined below. These topics are based on Helix interview data, although some areas, such as the importance of modeling and simulation, have been identified as important for the growth of the SE discipline in other sources as well (e.g. the *INCOSE Vision 2025* (INCOSE 2014)):

- **Balance and Optimization** – One of the critical challenges of systems engineers – and an area where systems engineers can provide substantial value – identified by the interviewees was the development of system balance and optimization. In particular, many interviewees stated that specialty engineers are often focused 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. The ability to maintain this balance is a critical skill for systems engineers. (Note: Proficiency in this topic is linked to proficiency in other areas, particularly in *SE Mindset and Interpersonal Communication*, both of which outline the skills necessary to support this type of approach.)

  Senior systems engineers in the Helix sample frequently cited this as a critical skill. They tend to focus their energies on the interactions of components and their impact on a subsystem's behavior and the interactions of subsystems and their impacts on the system's behavior more than the discipline engineers working on the components or subsystems. This focus gives the systems engineer the insight required to coordinate across disciplines, components, and subsystems to arrive at the optimal system solution. Often, this requires convincing discipline engineers that they do not need to make state-of-the-art components, but instead focus on components that will work best for the system overall. Convincing discipline engineers to "sub-optimize" some components also requires the abilities discussed in *Interpersonal Skills* and *Technical Leadership*.

  Junior systems engineers in the sample stated that understanding these interfaces and balance were important based on their training and education in SE. However, when asked about specific challenges they had faced, they tended to focus on technical challenges specific to a component and not on integration or systemic issues.

- **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 category represents and individual's ability to understand and perform modeling and simulation.

Systems engineers in the sample who stated they were proficient in modeling and simulation discussed thing like “whole system” modeling – the development of models meant to represent the overall system and not just independent aspects of a system. The simulations from these models, then, help the systems engineer to understand behaviors – expected and unexpected – at the system level. This often requires the integration of multiple types of models.

The junior systems engineers in the sample who discussed modeling and simulation generally talked about more specific aspects of modeling and focused on specific tool suites, such as MATLAB, Simulink, or models built in FORTRAN. Typically, these models were designed to represent a very specific aspect of the system. Junior systems engineers did not discuss the integration of these models with others to form a more holistic view of the system. They also generally discussed more about modeling and less about using the models to determine behavior (simulation).

- **Development Processes** – Each organization has its own processes that govern the development of systems. Many of the interviewees discussed the importance not only of understanding generic systems engineering processes, but understanding the specifics of “how systems are built here”. This includes understanding the organizational context and the processes and approaches that will be successful within that context.

  In general, the senior systems engineers stated that they had mastered the internal processes of their organizations, could successfully navigate these processes, and knew where the organization would allow leeway to ignore part of the process, and where it would not. Understanding the organizational context and business aspects of the processes is a critical aspect to the proficiency.

  Junior systems engineers in the sample generally stated that they understood the overall processes of the organization, but not necessarily why things were done in a certain way or how to accomplish systems engineering tasks within that process. Particularly, they used phrases such as “I need to learn how ‘x’ task is done here,” or “I don’t yet know who to call or who I’m not supposed to include in this task.” Generally, the junior systems engineers stated that mentors helped them to navigate these issues and begin to understand the organizational work processes.

- **Systems Engineering Tools** – Though the subject of tools was a common discussion during the Helix interviews, the most frequently-discussed types of tools were tools for requirements management – issues with and benefits of their use, access to them, etc. – and DOORS was the most frequently mentioned. This category represents an individual’s ability to utilize requirements management tools to support overall system development. A few interviewees discussed other types of tools, such as those that assist with project life management (PLM), though in general the discussion focused on the desire for and lack of availability for these tools.

  Senior systems engineers discussed the importance of having individuals in the workforce who can effectively use these tools. However, they themselves often did not use these tools directly. In general, they stated that they used the tools earlier in their careers, but as they progressed, others would handle those roles and they would focus on higher order issues. With regards to tools, senior systems engineers were generally concerned with the overall workforce access to and training in tools and the organization’s ability to provide support for these tools. In terms of their own use of these tools, they were concerned with the quality of the inputs and outputs and their ability to use the outputs to guide system development.

Junior systems engineers tended to focus more on the specifics of tools. For example, they might describe a shortcoming in a particular tool, but were less likely to discuss whether they believed their inputs were appropriate or the quality of the outputs. In other words, junior systems engineers were more concerned with the mechanics of the tools while senior systems engineers were more concerned with the strategic use of these tools.

Overall proficiency in *SE Methods, Processes, and Tools* is expected to be a combination of an individual's ability to work within and across the different topics outlined above. The mechanism for creating a proficiency level for the overall category, however, has yet to be determined, as the weighting of different topics may depend on the role being played. For example, a junior systems engineer may be assigned as a requirements database manager. That individual, therefore, would be expected to be very proficient in the systems engineering tools topic; however, he or she may not need to have a high level of proficiency in modeling and simulation. The inverse would be true for an individual assigned to work on a system model. And a senior systems engineer may need to use the outputs of these tools, but would not need to be an expert in these tools. Because of this, creating a single, generic weighting of these topics seems inappropriate. The Helix team will explore this further in future data collection.

In this category, education and training and experiences seem to play a fairly equal role in proficiency development, while mentoring was less often mentioned. See Section 5 for additional discussion.

4.4.4 CATEGORY: SYSTEM COMPLEXITY

System complexity is not currently broken into individual topics, but represents the degree to which a system's design or code is difficult to understand because of numerous components or relationships among components. (ISO/IEC 2009) In this context, the proficiency is the level of complexity an individual can handle. This may include things such as their ability to work on components, subsystem, systems, or systems of systems as well as the ability to work on products, platforms, services, or enterprises. At this point in time, Helix data does not provide much granularity on proficiency levels for complexity.

4.5 AREA: SYSTEMS ENGINEERING MINDSET

The fourth proficiency area is *Systems Engineering Mindset*. This area 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:

- **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.
- **Paradoxical Mindset** – The ability to hold and balance seemingly opposed views that are critical to providing value for systems engineers.
- **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 ability to take rational risks.
- **Inquisitive and Self-Driven** – The ability to ask the right questions without hesitation, and to self-initiate and complete work with little to no supervision or external impetus.

- **Quick Learning and Abstraction** – The ability to synthesize new information from separate pieces of data with multiple sources, e.g. to realize that a problem an electrical engineer faces in one component is related to a problem that a mechanical engineer faces in another component and address the root causes instead of just the symptoms.
- **Foresight and Vision** – The ability to foresee the remaining lifecycle of the system, the impact of current decisions, and to visualize possible scenarios.

The proficiency categories in *SE Mindset* and the subsequent areas of *Interpersonal Skills* and *Technical Leadership* may broadly be considered to be 'soft skills'. These are different from the more 'hard" or technically based skills that are included in the previous areas of *Math/Science/General Engineering*, *Systems Domain & Operational Context*, and *SE Discipline*. Development and evaluation of soft skills is addressed by the disciplines of psychology, social sciences, and management sciences. In those disciplines, proficiency levels of those soft skills are usually assessed using a Likert scale (Likert 1932) that may be featured in a questionnaire or in a survey. Evaluation of the proficiency levels of an individual is typically done by scoring the responses provided by that individual to a variety of questions along a Likert scale (Walumbwa et al. 2008, Gosling et al. 2003, MacKenzie et al. 2011). The items and number of levels in those scales vary, but they typically range from a strong negative rating to a strong positive rating.

Further, the level of proficiency of an individual in any of the hard skills would typically start from the lowest value and continuously progress towards the highest value. For example, a systems engineer could have typically started at the lowest level of proficiency in *SE Complexity*; but with learning and experience, he or she may have attained a reasonable level of proficiency while entering an organization as a junior systems engineer. This initial level would vary for different individuals, but in most cases, progress towards the highest level and actually attaining that would typically happen over the course of a career as a systems engineer.

In the case of soft skills, though, the starting level of proficiency need not be the lowest – individuals may have inherent abilities that are unrelated to their careers. Attaining the highest level is not dependent on seniority. For example, an individual could posses reasonable levels of proficiency in an *SE Mindset* category even while at high school, and could be close to the highest level when entering an organization as a systems engineer. Several senior systems engineers stated that when they learned what "big picture thinking" was, they realized that they had naturally viewed the world that way for their entire lives.

Based on all this, the Helix team has developed a simple Low/High scale to differentiate the proficiency levels in the *SE Mindset* area and the subsequent areas of *Interpersonal Skills* and *Technical Leadership*. This is influenced by the data – in general, people described what it meant to be highly proficient in these areas or to have low proficiency, but not much was discussed about the stages in between.

Table 5. *SE Mindset* Categories & Proficiency Levels

SE MINDSET	Low	High
'Big Picture' Thinking	• Has limited ability to think beyond the scope of the problem at hand	• Is able to think broadly across multiple dimensions
Paradoxical Mindset	• Is able hold one of two opposing concepts, but not both	• Is able to successfully hold both opposing concepts, and is also able to move from one to another as necessary

SE MINDSET	Low	High
Flexible Comfort Zone	• Is not confident enough able to expand beyond his or her comfort zone	• Is flexible and is able to permeate the boundaries of his or her comfort zone with ease
Inquisitive & Self Driven	• Hesitates to ask questions • Does not have the drive to take initiative; prefers to be a spectator	• Does not hesitate to ask 'why' questions in any gathering • Is able to initiate action by himself or herself
Quick Learning & Abstraction	• Is a relatively slow learner • Is unable to abstract lessons beyond the obvious	• Is able quickly grasp the critical bits of information • Is able to connect the dots at a higher abstract level
Foresight & Vision	• Is unable to see future effects and implications • Does not have a clear vision of the end-goal	• Is able to see future impacts • Has clear vision of the end-goal

The Helix team will explore this in future and work towards developing a richer scale for these soft skill categories.

Though interviewees discussed the roles of experiences, mentoring, and education and training in improving abilities in these areas, they also generally agreed that you had to have some level of innate ability. In other words, someone has to possess at least a minimal natural ability in these areas; in the opinions of the interviewees, they are not things that can be learned from scratch.

4.5.1 CATEGORY: BIG-PICTURE THINKING

Also referred to as 'holistic thinking' or 'systems thinking', big-picture thinking was very commonly identified as an important and essential characteristic of systems engineers. An interviewee said, "The difference between a systems engineer and a design engineer is that a design engineer doesn't have to see the big picture."

Helix participants used the term 'big-picture' to 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.

Many interviewees emphasized that a systems engineer was *the* person to bring this broader perspective, while specialty engineers and subject matter experts often tend to be narrowly focused on their area of interest. In many situations, systems engineers are not only called to provide this big-picture perspective, but to also enable others to see this bigger picture. Going along with the importance of innate abilities, however, several senior systems engineers indicated that one of the things they look for when recruiting systems engineers is a design engineer who indicates that they have this broader view.

4.5.2 CATEGORY: PARADOXICAL MINDSET

Another critical and unique characteristic of systems engineers that emerged from the Helix interviews was the ability to hold seemingly opposite perspectives simultaneously, and being able to move from one perspective to another appropriately – a paradoxical mindset. 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:

- **Big-Picture Thinking *and* Attention to Detail.** As discussed in the previous sub-section, 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. An interviewee elaborated, “There are some people who can see through a microscope and others who are more visionary. A good systems engineer has some elements of both”.
- **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.
- **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 together. 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).
- **Courageous *and* Humble.** Systems engineers need to be courageous as leaders; the proficiency area *Technical Leadership* discusses many elements of this. At the same time, they also need to be humble enough to recognize that there are others who are experts in their individual areas of specialization. They are confident in their abilities to make a tough judgment call with a ‘lack of pride’, but also willing to recognize and accept their mistakes.
- **Methodical *and* Creative.** 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 the problems at hand and arriving at solutions, without compromising the disciplined approach.

> There is a need for systems engineers to be flexible and adaptable, in order to effectively respond to change and unexpected disruptions.

4.5.3 CATEGORY: FLEXIBLE COMFORT ZONE

By definition, an expert possesses proficiency in a specific area, which is their ‘comfort zone’; and they typically have a “fear of going outside that circle or comfort zone”, as one interviewee said. Such experts provide value to the organization by contributing their expertise in those focused areas, and they prefer to stay that way – organizations surely need such individuals. 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. One interviewee said, “[Systems engineers] have the inability to be pigeon-holed.”

It is common in complex system development to encounter areas within the problem and solution domains that do not fall within traditional boundaries of expertise and specialization. Even if they did, interactions between the different areas of specialization could be unfamiliar territory; such unfamiliarity could also bring in ambiguity and uncertainty. Systems engineers tend to embrace such unfamiliar situations with reasonable ease. In the early development of systems engineers, one interviewee stated that some fresh graduates, in particular, need to be “forced to get out from their comfort zones”; “some people push themselves, and some people need to be pushed”, he said.

4.5.4 CATEGORY: INQUISITIVE AND SELF-DRIVEN

Effective systems engineers possess a high level of inherent curiosity. They always want to know more and have a ‘hunger for knowledge’. They also do not hesitate to ask questions. Asking questions is not a sign of weakness, but it is important to ask the right questions clarified one interviewee. “You are not always the stupidest person to ask questions; you are just the bravest one to ask the questions”, he said. “Everyone assumes everyone knows”, said another interviewee; asking questions ensures clarity and forces a shared common understanding.

Flowing from this curiosity is the self-drive that systems engineers possess. They are not satisfied with just answers to their questions, but are proactive and highly energetic to initiate and make things happen. One interviewee elaborated: “Initiative is a major characteristic of a systems engineer. It separates the specialty guys from the systems engineer. The systems engineer can not wait for all of these requirements to be in a nice ball and put down the pipeline. In a lot of situations, the systems engineer has to start taking the initiative to put people and ideas together.”

Systems engineers have the ability not only to initiate, but to learn, gather resources, and to perform required tasks without the needs for guidance, reference, or hand-holding.

4.5.5 CATEGORY: QUICK LEARNING AND ABSTRACTION

In many situations, systems engineers “don’t have the time to understand everything”; they need the ability to “pick out critical bits of information, at the right level.” In some cases, there could even be an information deluge, where systems engineers need to quickly filter out what is not required. And even with that filtered information, systems engineers need to know “when to use or not use pieces of information”, said one interviewee. Another interviewee said that systems engineers “understand things on the fly”, and make inferences.

Beyond just being a quick learner, systems engineers need the ability to abstract the core of the information, and internalize it. Doing so enables them to “stay above the noise” and to quickly identify “obvious pitfalls and obvious opportunities.” 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.

4.5.6 CATEGORY: FORESIGHT AND VISION

Another important ability for systems engineers is to be able to foresee situations and scenarios, and to mentally simulate them. Every decision or change is likely to have an impact beyond the current confines of time or space. Systems engineers need to have foresight to evaluate the impacts to help guide effective decision-making. 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.

Along with foresight is the ability to remain focused on the end vision of the system, and to drive individuals, teams, and their activities towards achieving that end vision which may not be easily perceivable.

4.6 AREA: 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 people. Irrespective of any formal leadership roles they may or may not have, 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, and further elaborated in subsequent sub-sections:

- **Communication** – Communication is key for systems engineers, and is a broad category covering a wide variety of related skills and abilities.
- **Listening and Comprehension** – The ability to listen to others' points of views and perspectives, and to comprehend and internalize the message accurately.
- **Working in a Team** – The ability to work collaboratively in a team.
- **Influence, Persuasion, and Negotiation** – The combined ability to 'make a point', and to successfully obtain buy-in from others on that point.
- **Building a Social Network** – The ability to build a network of professionals that can serve as a resource that may be tapped into.

As discussed in Section 4.5 for the SE Mindset area, a simple Low/High scale is used to differentiate the proficiency levels in this area, as shown in Table 6.

Table 6. *Interpersonal Skills Categories* & Proficiency Levels

INTERPERSONAL SKILLS	Low	High
Communication	• Is able to communicate only to one type of audience • Is able to communicate only one type of content • Is able to communicate using one mode of communication	• Is able to communicate to different audiences • Is able to communicate different types of content • Is able to communicate using multiple modes
Listening and Comprehension	• Is not a patient listener • Does not fully comprehend what is said	• Is a patient listener • Fully comprehends what is said
Working in a Team	• Is more comfortable working alone	• Is comfortable working in teams and is a good team player
Influence, Persuasion and Negotiation	• Unable to successfully obtain buy-in	• Is able to obtain buy-in
Building a Social Network	• Has a limited social network to tap into	• Has a large social network to tap into

The Helix team will focus on understanding the progression between "low" and "high" proficiency in Interpersonal Skills going forward.

4.6.1 CATEGORY: COMMUNICATION

Systems engineers interact with a variety of people. Often the 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. Therefore, communication skills are foremost, and comprise many related skills and abilities, as discussed below.

- **Audience** – Systems engineers need to communicate with a variety of direct and indirect audiences. One interviewee stated that requirements are what the systems engineer uses to communicate "what the product needs to be" to the development team; another interviewee stated that systems engineers need to communicate with upper management and that the message could go all the way to the Office of the Secretary of Defense and even to Congress. Systems engineers communicate with a variety of audiences including customers, subject matter experts, program managers, vice presidents, directors specialty engineers, problem owners, technical teams, contractors, decision makers, and system testers and others working on the project. Depending on the audience, the content, mode, and style of communication needs to be tailored. To some audiences, much technical detail and elaboration needs to be provided; to others, the message needs to be relayed in plain language with minimal technical jargon.

- **Content** – The variety of content that systems engineers need to communicate can be broadly divided into three types:

  - **Technical**: Communications with disciplinary and specialty engineers and subject matter experts involve high technical content; 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 content.

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

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

- **Mode** – Communicating the intended content to the target audience is done through a number of different modes, and systems engineers need the skills to effectively communicate in these multiple modes. Some interviewees observed that by nature, specialty engineers tend to be poor communicators in all modes; to be effective, systems engineers need to develop their proficiency in these areas:

  - **Oral**: Oral communication 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. Systems engineers need the ability to clearly express their thoughts and perspectives to establish a shared common understanding.

  - **Presentation**: A special form of communications is the ability to stand in front of an audience and to deliver a presentation using appropriate aids. Effective presentations, particularly to stakeholders, help maintain a shared understanding of a system and enable stakeholders to make decisions. Presentations are more than just talking in front of an audience – "you have to defend your position; you have to be technically sound; you cannot afford to make any large mistakes". 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.

  - **Writing and Documentation**: Written communication skills are equally critical for systems engineers. One of the interviewees described the culture in a particular organization, where the boss would say, "Don't call me up; write to me. If you call me, it means you probably haven't thought through it, and I don't have time. If you write to me, you've sat down, figured it out, and took time to communicate it efficiently. Then I probably have time to talk to you." While we can appreciate the rationale of this boss, all writing does not automatically become effective; it requires skill and effort. 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 a comprehensive design document being submitted for a Critical Design Review. It must be noted that in some organizations, systems engineers are primarily viewed as reviewers and keepers of documentation.

4.6.2 CATEGORY: LISTENING AND COMPREHENSION

Although this may be considered part of communication skills, listening is being called out as a separate category. One of the interviewees pointed out that “[Listening] is one thing that engineers don’t do a good job at. We tend to interrupt each other; we want to get our opinion across.” For systems engineers, listening begins at 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. Also, it is not passive listening and just allowing someone to talk; it is active listening with an intent to understand and comprehend what is being said – such listening would include probing and engaging in a conversation.

Listening may not be easy to do in some situations. As one interviewee said, “its challenging at times to be a good listener, and to take criticism well, and to still be able to argue your point without trying to be confrontational or intimidating or defensive.”

4.6.3 CATEGORY: WORKING IN A TEAM

“Systems engineering is an extreme team sport,” said one interviewee. 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. It is typical for an organization to perceive its system engineering capability as the collective capability of its systems engineering workforce and not just as the sum of the capabilities of its individual systems engineers. Hence, team dynamics and synergy are key to the functioning of a systems engineer. To be able to work effectively in a team is not only about contributing as a member of the team but also about allowing others to contribute and respecting their perspectives. Conflicts and differences of opinion are bound to arise, but it is important to resolve them in an objective, impersonal manner.

4.6.4 CATEGORY: INFLUENCE, PERSUASION, AND NEGOTIATION

Although this may be considered as an aspect of leadership, it is critical for every systems engineer to have the skills needed to make a point and to successfully obtain buy-in. In many situations, systems engineers bring in a perspective that is different from what others may bring in: a focus on the overall system, and on customer’s needs. In such situations, it requires influence, persuasion, and negotiation skills for systems engineers to enable others to see the bigger picture on which they need to focus. As one interviewee stated humorously, “You don’t have the ability to just go in and hit them with a stick and say ‘you’ll do it’; or even if you had that ability, it’s not particularly effective.”

4.6.5 CATEGORY: BUILDING A SOCIAL NETWORK

A systems engineer needs to be a ‘people person’ not just to be a better communicator, but to also build a social network of professional acquaintances. Such a network becomes a valuable resource for systems engineers 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.

4.7 AREA: TECHNICAL LEADERSHIP

The sixth and final *Atlas* proficiency area is *Technical Leadership*. “I believe that systems engineers are the ones who get all of the other engineers to work together,” said one interviewee; “systems engineers tend to gravitate towards leadership,” said another. It is common and natural for systems engineers to play leadership roles at many levels within an organization, and this aspect was frequently discussed during Helix interviews.

The specific categories contained within *Technical Leadership* are listed below, and further elaborated in subsequent sub-sections:

- **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.

- **Balanced Decision Making and Risk Taking** – This includes the ability to make sound decisions considering a variety of factors, constraints, perspectives, and objectives. It also includes taking rational risks.

- **Managing Stakeholders and their Needs** – This includes the ability to manage all the internal and external stakeholders, and keeping the team focuses on their needs, especially those of the end user or customer.

- **Conflict Resolution and Barrier Breaking** - This includes the ability to resolve conflict within the team as well as between other individuals and groups, keeping the system goals in mind. This also includes breaking barriers that prevent progress towards completion and success.

- **Business and Project Management**– The ability to handle a variety of business and project management activities including accounting, budget, cost estimation, schedule, work breakdown, and profit.

As discussed in Sections 4.5 and 4.6, a simple Low/High scale is used to differentiate proficiency levels in this area.

Table 7. *Technical Leadership* Categories & Proficiency Levels

TECHNICAL LEADERSHIP	Low	High
Building and Orchestrating a Diverse Team	• Is unable to build the right team for the job • Is unable to guide a team to success	• Is able to build the right team for the job • Is able to guide a team to success
Balanced Decision Making and Risk Taking	• Takes decisions considering only limited perspectives • Hesitates to take any risk	• Takes decision considering all perspectives • Is confident to take rational risks
Managing Stakeholders and their Needs	• Is able to manage only few stakeholders and some of their needs	• Is able to manage many stakeholders and their needs
Conflict Resolution and Barrier Breaking	• Is unable to successfully resolve conflict	• Is able to successfully resolve conflict
Business and Project Management Skills	• Is unable to employ business and PM skills	• Is able to employ business and PM skills

The Helix team will focus on understanding the progression between "low" and "high" proficiency in Technical Leadership going forward.

4.7.1 CATEGORY: BUILDING AND ORCHESTRATING A DIVERSE TEAM

One of the interviewees said, "If you develop the right people on the right team, you will get the right results." 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. In some cases, the systems engineer is able to select and gather the required team; while in other cases, the systems engineer is given the team that needs to do the job.

Building and developing an effective team requires the systems engineer to fully know each of the team members: their strengths, weaknesses, capacities, capabilities, limitations, personalities, expertise, and working styles. This awareness will enable the leader to identify gaps that need to be addressed, and to think about how to "play them in a way that as a team they are more effective." In addition, the leader also needs to know what it takes for the team members to do their jobs. Going forward, *Atlas* may assist systems engineers in understanding their team's capabilities.

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. An interviewee used an old adage to say that, "It is better to 'teach a man to fish' than to give him fish."

The systems engineer needs to establish good relations with and among the team members and to make sure that everyone has a voice – not just the strongest or the loudest person. It is also important as a leader, to create an atmosphere of mutual respect amongst the team; to help them remain focused; and to ensure that they have proper tools in order to achieve results. The systems engineer needs to ensure that the entire team shares a common understanding of the problem to be solved and the ultimate vision for the solution.

Another key aspect of handling a team is the ability to delegate. It was observed by an interviewee that, "Sometimes systems engineers tend to do the work themselves because they love doing the work." The leader needs to build enough trust in the team to be able to delegate with confidence.

4.7.2 CATEGORY: BALANCED DECISION MAKING AND RISK TAKING

"There are many who point to problems, but [systems engineers] 'own' the problem," said one interviewee. Solving the problem requires the leader to take a number of balanced decisions along the way. This decision-making needs to be structured and balanced, considering all relevant variables. The challenge is that most often all the required information may not be available and the leader needs to have the ability to gather any additional information and to take a decision based on even insufficient information.

The ability to make such decisions requires the leader to be comfortable in dealing with ambiguity and uncertainty and to be able to take rational, calculated risks. A high proficiency in *SE Mindset* would provide much of the ability and rationale to consider multiple options and to then take a logical, rational decision. The additional challenge is that in most cases, the priorities and variables are likely to keep changing.

The systems engineer also needs the ability to make such decisions quickly. One interviewee noted that some people "will circle on the same problem forever if you let them." At the same time, the systems engineer must not "jump to a solution and hope that it meets the customer's requirements." In addition to making the judgment call in making those decisions, the systems engineer also needs the ability to enable others, particularly subject matter experts and other engineers, to make such decisions.

In all this decision-making, the systems engineer needs to understand the implications of these decisions and their scope of impact; and also be aware of the risks and possible means to mitigate them.

4.7.3 CATEGORY: MANAGING STAKEHOLDERS AND THEIR NEEDS

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.

Managing internal stakeholders requires an understanding of the structure and functioning of the parent organization and its leadership. Internal stakeholders also include disciplinary experts and other internal groups. Similarly, external stakeholders would include customers and organizations with their own cultures and characteristics.

While interacting with the stakeholders, the systems engineer needs to "understand the motivation of the guys across the table." Therefore, the systems engineer also becomes the person who needs to be aware and mindful of the needs and expectations of each of those stakeholders. Conflicting needs and expectations need to be resolved, and the focus of the system development must be maintained on the vision of the customer and making sure the end product is what the customer expects.

In addition to making the required decisions, the systems engineer needs to know how to respond to management, customers, and all other stakeholders.

4.7.4 CATEGORY: CONFLICT RESOLUTION AND BARRIER BREAKING

"People don't all think the same; and they don't all talk the same," said one interviewee. Conflicts are bound to rise in a variety of scenarios, and as a leader, it is upon the systems engineer to resolve these conflicts. Also, systems engineers serve as the link between the technical side and business side of the organization, and conflicts could arise between these two sides as well. It is important that these get resolved quickly and effectively, because "you can't go forward until everyone is coming along with you; you don't just leave people behind."

The systems engineer needs the ability to gather consensus, help people compromise, and thereby resolve the conflict. To do so, the leader needs to know where the conflict really exists, which team members or stakeholders need to talk to each other, and to facilitate an effective dialogue between them. Facilitating this dialogue is more than just letting them talk to each other, but also understanding their conversation so that the systems engineer can intervene, explain, and provide clarity as required. In doing this, the systems engineer needs to be able to disengage emotionally. This requires the systems engineer to effectively deal with stress.

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. Related to this is also the ability to shield engineers and other team members from organizational changes and disruptions, to the extent possible.

4.7.5 BUSINESS AND MANAGEMENT SKILLS

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 be proficient in business and management skills. A systems engineer is also expected to be comfortable in management positions.

Though most system requirements could directly relate to technical aspects, the systems engineer must not forget the business side of requirements. One of the interviewees explained that "an effective systems engineer is someone who just remembers ultimately what the objective is: that 'I need to be solving the problems of customers in a way that the customers are willing to buy it'." This is a business decision by the customer.

Some of the decisions that a systems engineer takes or facilitates could be business or management related decisions, such as fiscal decisions. However, there is likely to be a business impact of every technical decision that is taken. The systems engineer must be cognizant of this.

The systems engineer is expected to be good in cost, accounting, schedule, work breakdown structure, work allocation, planning, and coordination. Systems engineers also need to be organized and good in time management: managing their own time, as well that of others in the team.

4.8 WAY AHEAD FOR PROFICIENCY

In 2015, the Helix team will begin validating the current proficiency model with new interviewees as well as in follow up interviews, focusing on validation of the areas, categories, and topics. The team will also work on fleshing out proficiency criteria that will make the model more implementable. In particular, the team will explore how proficiency in a set of topics would combine to create a proficiency level for a given category and likewise rolling up category proficiencies for a given area.

5. FORCES THAT IMPACT THE PROFICIENCY OF SYSTEMS ENGINEERS

As outlined in Figure 1, the Helix team has identified forces that impact the proficiency of systems engineers. This section will elaborate on the primary forces that may influence systems engineers' proficiency and define in what ways specific forces may influence specific proficiencies. In addition there are both organizational and personal characteristics that will influence an individual's proficiency. These variables are shown in Figure 6.

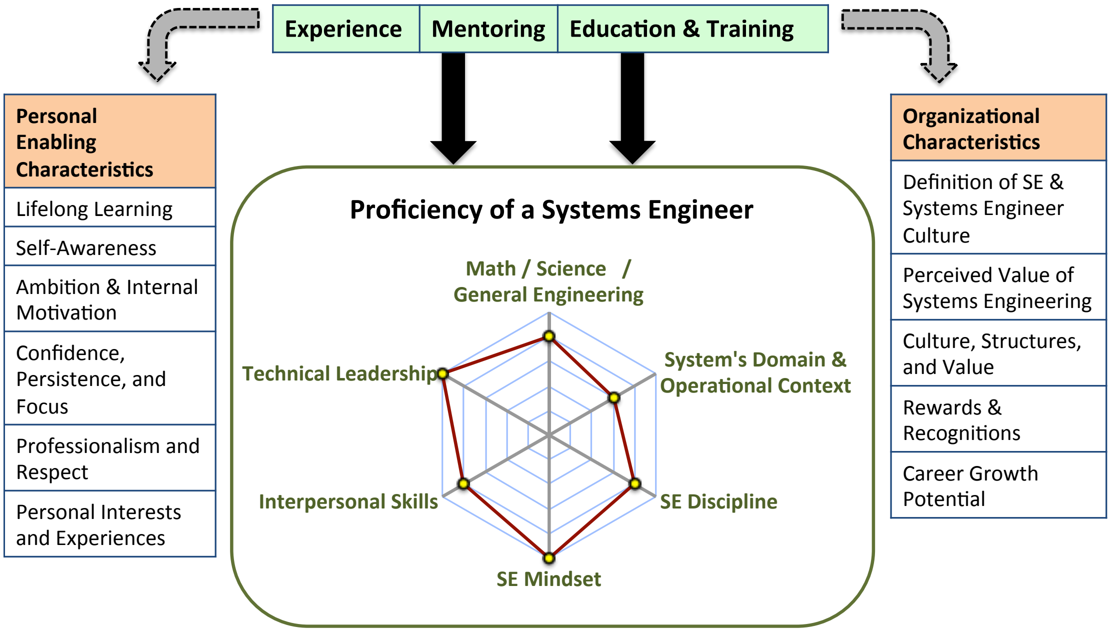


Figure 6. Forces that Impact Proficiency

The Helix team has identified three forces that impact the proficiency of a systems engineer:

- **Experiences** – In this context, experiences are specifically the activities an individual has participated in – either as an observer, performer, or leader – in a professional setting.

- **Mentoring** – A professional relationship in which an experienced person assists another in developing specific skills and knowledge that will enhance his or her professional and personal growth. (Management Mentors 2014)

- **Education and Training**. A structured mechanism for transferring desired information, skills, or behaviors to an individual or group of individuals in an academic setting. In the context of Helix, education is generally considered to be more formal and usually refers to courses that cover more generally-applicable knowledge and lead towards a degree or certificate. Training is often less formal, structured around a specific topic, and tailored to a particular environment (organization, domain, etc.).

How these primary forces affect proficiency of systems engineers is elaborated in the sections below.

5.1 EXPERIENCES

As one interviewee stated, "systems engineering is a contact sport" – understanding of the discipline requires practice in a real-world setting. This is a widely agreed point within the community. In fact, every single Helix interviewee has stated in some way that experiences are a critical component to the development of systems engineers. One of the debates in the community, however, has been around the length of experiences and which types of experiences are really necessary to grow a systems engineer. As discussed in earlier Helix reports (Pyster et al. 2013 and 2014), at the high level there are several areas where Helix interviewees strongly agree:

- Experiences in different parts of the systems lifecycle are useful because they enable a systems engineer to "see" or "feel" the consequences of decisions and how they play out in different phases of the lifecycle.
- Experiences with different types of lifecycles (e.g. Quick Reaction Capability (QRC), formal acquisition lifecycles, aligning with DoD 5000.2; and internal research and development (IR&D) projects) are useful because they provide the prospective necessary to enable process tailoring.
- Experiences on different aspects of a system (e.g. part, component, subsystem, system) and different critical orthogonal attributes of the system (e.g. weight, size, etc.) also enable development of pattern recognition that may enable individuals to more readily predict and respond to challenges.
- Failure experiences - experiences where technical, process, or organizational aspects of a system do not run according to plan – are often cited as slightly more valuable than successful experiences.

It is impossible to take each of these as a standalone recommendation for the development of systems engineers, however. For example, with regard to experiences across different aspects of the lifecycle, interviewees stated that there are both benefits and drawbacks to experiencing the lifecycle across one system versus experiencing aspects of the lifecycle on multiple systems. One interviewee stated that ownership and internalization of lessons learned is not the same when an individual moves between projects; in his words, "you don't get that emotional punch to the gut," when you deal with a bad decision that was made by someone else. The counter to this is that defense systems sometimes have incredibly long lifecycles; allowing a systems engineer to grow up within one system would often take many years and is not conducive to more quickly growing the workforce.

The Helix team will continue to explore these more general findings on experiences going forward, with particular focus on developing a metric to determine the quality of these experiences. The rest of this section, however, will focus on the alignment between experiences and the proficiency areas discussed in Section 4. This discussion is intentionally kept at a fairly high level.

Specific considerations for how experiences may impact an individual's proficiency level from the Helix interviews are discussed in Table 8, below.

Table 8. Experiences that impact proficiency levels.

Proficiency Area	Category	Experiences
Math/Science /General Engineering	Probability & Statistics	• In general, experiences were not mentioned in conjunction with proficiency in math, science, or general engineering principles. A few people made statements to the effect that practical experience was important for really internalizing some of the fundamentals of engineering, but beyond that, no specifics were provided. This is reasonable, given the somewhat theoretical nature of the categories contained within this area.
	Calculus & Analytical Geometry	
	Natural Science Foundations	
	Engineering Fundamentals	
	Computing Fundamentals	
System's Domain & Operational Context	Relevant Domains	• Two types of experiences were mentioned that would build proficiency in the relevant domains for a given system: • Experiences working in the primary domain for a system. This is a simple digital translation – either an individual has worked in the primary domain or has not. • Experiences working in related domains. This is a slightly more nebulous idea, as many other domains could be considered relevant. For example, an individual may not have worked on a space system, but may have worked on an aeronautical system, where many of the same physical forces and principles will be applicable. • Whether an individual has worked in the primary or related domains, another consideration is the depth and breadth of those experiences. For example, did they see across the entire lifecycle in a given domain, or did they work on one narrow aspect of the lifecycle? Did they work on multiple systems – therefore increasing their depth of understanding in the domain – or only one?
	Relevant Technologies and Systems	• For any given system, a suite of technologies or enabling systems may be relevant. Experiences working with these technologies improve a systems engineer's overall grasp of the technical challenges and opportunities for a system. These experiences also help to create detailed understanding of some aspects of the system, which may free the systems engineer to focus on the "big picture".
	Relevant Disciplines	• The majority of senior systems engineers in the sample state that it was critical for a systems engineer to have a grounding in a technical discipline; the most commonly cited were electrical, mechanical, or software engineering. • For any given system, some disciplines will be more relevant than others. Relevant disciplines also change depending on the type of system – product, service, or enterprise. For example, for a radar system, electrical engineering will be critical, but for a service system, electrical engineering may be far less critical. • As with domain, experiences working in the most critical disciplines for a given system provide a direct correlation with proficiency for working on that system. If an individual has no relevant experiences, they will begin working on the system as a novice. If they do, then the character– again, the activities, lifecycle phases,

Proficiency Area	Category	Experiences
		etc., experienced – and quality of those experiences – will determine an individual's proficiency level.
	Familiarity with the System's CONOPS	• There are a few aspects of experiences that may be relevant here. The first is familiarity with the specific CONOPS for that system. For example, someone with experience of helping to develop that CONOPS will be much more proficient than someone who is simply provided some documentation of that CONOPS to review. • The second aspect is familiarity with related CONOPS. So, for example, an individual may have experiences working on systems with a similar CONOPS, and would understand in practice some of the constraints and challenges that may arise. • Experiences working in the operating environment may also be very valuable. Interviewees with prior military experience, for example, stated that though they had not worked on the design of systems, experiencing how these systems actually operate in the field and having first-hand experience with the environment provided critical insight that helped them make better decisions about design.

Systems Engineering Discipline	Lifecycle	• As discussed above, the link between experiences across a variety of lifecycle phases and effectiveness became an early finding for Helix. There are several aspects of lifecycle experiences to be considered:
		  • The specific phases of the lifecycle that an individual has experienced. This does not necessarily have to be the current phase of the lifecycle; as discussed above, experiences in the later phase of the lifecycle may help an individual make better decisions earlier in the lifecycle.
		  • The length of time spent in different phases of the lifecycle. For example, an individual who did a six month rotation in testing, a six month rotation in manufacturing, then spent the next ten years working on requirements will not have the same overall proficiency as an individual who has spent a few years in each of these activities.
		  • The breadth of systems worked on for each aspect of the lifecycle; these may be experiences in different domains or technologies, or a suite of similar systems.
		  • Experiences actually controlling or tailoring the lifecycle process.
		• The Helix team will explore the balance of depth and breadth for lifecycle proficiency going forward. Based on current data, the "desired" state is dependent on the role an individual is playing.

	Systems Engineering Management	• Experiences performing any of the orthogonal systems engineering functions. Time for each experience and the diversity of these experiences will impact proficiency. As with lifecycles, experiencing one aspect of SE management, e.g. planning, on multiple systems or across technologies or domains, gives the individual a better opportunity for pattern recognition while working more narrowly

Proficiency Area	**Category**	**Experiences**
		gives the opportunity to develop focused expertise. • Overall, the current Helix data is light on discussion of SE management activities; collection around these areas will be a focus going forward.
	Systems Engineering Methods, Processes, and Tools	• Experiences working on specific tools were discussed by some interviewees. It was noted that the need for tool-focused proficiency is generally higher earlier in one's career; as one progresses, the need to understand the outputs and limitations of these tools becomes a more important proficiency than the ability to do the mechanics. • While the importance of these areas was mentioned in many interviews, few specific experiences that help to improve proficiency were discussed relating specifically to methods, processes, and tools.
	System Complexity	• Experiences working on different levels of a system – component, subsystem, system, system-of-systems – will impact an individual's proficiency. Someone who has worked only on components will not be expected to be as proficient as someone who has had system-level responsibilities. • Responsibilities specifically related to integration may be important for complexity, as they force an individual to look outside his or her assigned system element.
Systems Engineering Mindset	Big-Picture Thinking	• The current Helix data does not include many insights into the role of experiences on the development of the *Systems Engineering Mindset*. • In terms of workforce development, some senior systems engineers have stated that when they are looking to identify new systems engineers, they do so through demonstration of these qualities in experiences.
	Paradoxical Mindset	
	Flexible Comfort Zone	
	Inquisitive and Self-Driven	
	Quick Learning & Abstraction	
	Foresight & Vision	
Interpersonal Skills	Communication	• Experiences working on teams were the primary mechanisms discussed for gaining proficiency in the *Interpersonal Skills* area. This diversity can be from either working on many different teams or by working on a team with diverse individuals. • Where discussed, the recommendation to work on "diverse" teams could mean different engineering disciplines, domain expertise, degrees of seniority or even teams with more balanced gender representation or ethnic diversity. • Many of the experiences shared during interviews were failure experiences – areas where failures in interpersonal skills led to problems with a system and highlighted the criticality of these skills.
	Listening & Comprehension	
	Working in a Team	
	Influence, Persuasion, & Negotiation	
	Building a Social Network	
Technical Leadership	Building and Orchestrating a Diverse Team	• Experiences leading teams of any size were discussed as being important to the development of systems engineers. • Experiences making decisions in different environments, particularly under uncertainty, were discussed as critical for
	Balanced Decision Making and Risk Taking	

Proficiency Area	Category	Experiences
	Managing Stakeholders and their Needs	systems engineers. Several interviewees stated that these experiences are particularly helpful when an individual can "feel" the impact of the decisions he or she makes.
	Conflict Resolution and Barrier Breaking	
	Business and Management Skills	

The Helix team will continue to collect data on experiences going forward and will focus some effort on exploring the particular experiences that support proficiency in the *SE Mindset*, *Interpersonal Skills*, and *Technical Leadership* areas, as these are currently the least mentioned in the current data.

Currently, there is a catalogue of many different types of experiences discussed by the interviewees and the specific lessons learned or values gained from those experiences – these are both retrospective (what was done and learned) and prospective (what one hopes to do and what one hopes to gain from it). The Helix team will continue to examine this data to develop experience patterns that may further inform the proficiency model for *Atlas v. 0.5*.

5.2 MENTORING

During the interviews, participants various organizational initiatives aimed at improving the efficiency of systems engineers, and the most commonly discussed initiative was mentoring. There are formal and in-formal arrangements in practice, depending on the level of organizational involvement in establishing and sustaining a mentoring initiative; participants offered their perspectives on the strengths and weaknesses of either arrangement.

In any typical mentoring arrangement, there is a mentor who 'gives', and a mentee who 'receives'. This arrangement is most beneficial to the mentee, but there are some benefits to the mentor as well. Indirectly, the organization also stands to benefit.

The focus of the discussions here is on mentoring as a force, that helps increase the level of proficiency of systems engineers. Table 9 describes ways in which mentoring may impact proficiency levels in the areas and categories identified in the proficiency model

Table 9. Mentoring impacts on proficiency levels.

Proficiency Area	Category	Mentoring
Math/Science /General Engineering	Probability & Statistics	• Mentoring plays a minimal role in developing the proficiencies in this area. Most of the foundational concepts in Math/Science/General Engineering are obtained from formal education and other related training.
	Calculus & Analytical Geometry	
	Natural Science Foundations	
	Engineering Fundamentals	
	Computing Fundamentals	
System's Domain &	Relevant Domains	• Mentoring plays a significant role in this proficiency area. • Systems engineers need to deal with a variety of domains,
	Relevant Technologies and Systems	

Proficiency Area	Category	Mentoring
Operational Context	Relevant Disciplines	technologies and systems, and they may not have the relevant education background or direct experience. In such situations, mentoring enables quick learning; and combined with experiences, systems engineers are able to increase their level of proficiency more effectively.
	Familiarity with the System's CONOPS	
Systems Engineering Discipline	Lifecycle	• Learning through a variety of experiences appears to be the typical way to improve the effectiveness of a systems engineer in the SE discipline related categories. Junior systems engineers, typically watch or support senior engineers do their jobs. Though much can be learned by observing, active mentoring significantly increases the learning and thus develops systems engineers more effectively. • Most organizations establish internal processes and practices for performing systems engineering activities. Mentoring enables systems engineers to understand those processes and practices better.
	Systems Engineering Process	
	Systems Engineering Management	
	Systems Engineering Methods, Processes, and Tools	
	System Complexity	
Systems Engineering Mindset	Big-Picture Thinking	• Unlike the proficiency areas discussed above, an individual may possess high levels of proficiency in the *SE Mindset* categories from a very early age. However, the ability to apply these to systems engineering related activities on the job may be limited. • Close association with a mentor and observing how the mentor functions, is very valuable to the mentee. • In addition, the mentor can elaborate the rationale and thinking behind some of his or her decisions and external behavior. This would greatly help the mentee.
	Paradoxical Mindset	
	Flexible Comfort Zone	
	Inquisitive and Self-Driven	
	Quick Learning & Abstraction	
	Foresight & Vision	
Interpersonal Skills	Communication	• *Interpersonal skills* is an area where mentoring can benefit the mentor as well. While a senior systems engineer may have already sharpened his or her interpersonal skills with many years of practice and experience, interacting with a mentee is different from interacting with stakeholders, customers and subject matter experts. • Strong communication skills are required by the mentor to successfully communicate the intended message to the mentee. • The mentee benefits greatly by observing and obtaining guidance from the mentor on various aspects of communication • In most cases, when there is a good relationship between the mentor and the mentee, the social network of the mentor eventually becomes the social network of the mentee.
	Listening & Comprehension	
	Working in a Team	
	Influence, Persuasion, & Negotiation	
	Building a Social Network	
Technical Leadership	Building and Orchestrating a Diverse Team	• Systems engineers spend much time being part of a team, before they become senior enough to lead a team. Hence, much of the proficiencies in *Technical Leadership* are first learned by being lead by an effective leader. • While in a team, the systems engineer is primarily the beneficiary of good leadership. Those experiences themselves provide valuable lessons. In addition, if the leader were also the mentor, then that significantly impacts the development of these proficiencies in the systems engineer.
	Balanced Decision Making and Risk Taking	
	Managing Stakeholders and their Needs	
	Conflict Resolution and Barrier Breaking	
	Business and	

Proficiency Area	Category	Mentoring
	Management Skills	

The above table summarizes the benefits of mentoring, where it acts as a force that helps increase the proficiency level of systems engineers in the many areas and categories; mentoring has also helped develop or strengthen some of the personal enabling characteristics. Also, the focus of the mentoring that was provided or received by most Helix participants was not limited to systems engineering; many mentees have benefited much from career guidance and 'critical life lessons' they received from their mentors. Mentoring is not just about pairing a mentor and a mentee, but rather a building of a potentially long relationship. Hence, various factors related to how the relationship is established, nourished, and sustained, determine the success or failure of a mentoring arrangement.

5.3 EDUCATION AND TRAINING

Education and training include any structured mechanisms for transferring desired information, skills, or behaviors to an individual or group of individuals in an academic – i.e. not "real world" – setting. While interviewees almost unanimously agreed that education and training had a role in the development of systems engineers, it was seldom "top of mind". All interviewees discussed experiences, most mentoring; about a third of the individuals who discussed education and training did so only after prompting by the Helix team.

Table 10 describes the ways in which education and training may impact proficiency levels as discussed in the current Helix data.

Table 10. Education and training that impacts proficiency levels.

Proficiency Area	Category	Education & Training
Math/Science /General Engineering	Probability & Statistics	• When discussed, interviewees stated that foundational concepts in math, science, and general engineering were primarily started through undergraduate education in traditional engineering disciplines; e.g. electrical, mechanical, civil, etc. • Some interviewees believed that a master's level education in a traditional engineering discipline was very beneficial to becoming extremely proficient in this area.
	Calculus & Analytical Geometry	
	Natural Science Foundations	
	Engineering Fundamentals	
	Computing Fundamentals	
System's Domain & Operational Context	Relevant Domains	• Development of proficiency in *System's Domain & Operational Context* was almost exclusively discussed as experience and mentorship based. • A few individuals noted that organizations may provide training regarding specific domains or technologies that are of particular importance for the business. When asked about the efficacy of this training, individuals stated that this provided a good overview or basic working knowledge, but nothing more. This would generally relate to a "novice" level of proficiency.
	Relevant Technologies and Systems	
	Relevant Disciplines	
	Familiarity with the System's CONOPS	
Systems Engineering	Lifecycle	• Internal organizational training that covers the systems engineering lifecycle and lifecycle considerations is sometimes available.

Proficiency Area	Category	Education & Training
Discipline		Generally, this is packaged as an "introduction to systems engineering' module. Training modules discussed by interviewees ranged from 0.5 days to two weeks in length. Where available, these were generally agreed to be a "good overview", often seen as helpful to young systems engineers but generally less so for senior systems engineers. • Where individuals pursued a master's degree in systems engineering, they indicated that this was very helpful to understanding the lifecycle and at least in a theoretical way understanding the impact of each lifecycle phase on subsequent phases.
	Systems Engineering Management	• Education and training for this category was not generally discussed by the interviewees.
	Systems Engineering Methods, Processes, and Tools	• This is a category for which training seems to be more readily available. Specifically: ◦ Organizations often offer training on their specific processes and methods that at least provide a general overview (though some provide some very detailed information) ◦ Organizations often offer training on their most commonly-used tools. These training modules provide at least a working knowledge, often with some hands on practice. • Where individuals pursued a master's degree in systems engineering, they indicated that an improved understanding of methods and processes were generally valuable.
	System Complexity	• No training or education relating specifically to system complexity was discussed.
Systems Engineering Mindset	Big-Picture Thinking	• Very little was discussed in terms of education and training for *Systems Engineering Mindset*. • A few individuals stated that pursuit of a master's degree in systems engineering helped them to improve their skills related to systems thinking or paradoxical mindset. However, they also stated that having real-world experiences on which to apply these principles made the learning more effective. • A few organizations provide training on systems thinking; feedback on this training ranged from very helpful to very ineffective.
	Paradoxical Mindset	
	Flexible Comfort Zone	
	Inquisitive and Self-Driven	
	Quick Learning & Abstraction	
	Foresight & Vision	
Interpersonal Skills	Communication	• There were very few examples in the data of training or education for interpersonal skills. • A few individuals indicated that training for conflict resolution, listening skills, or communication were available through their organizations. Where available, individuals indicated that this provided some useful principles and awareness, but that this information had to be put into practice to be internalize and made more useful.
	Listening & Comprehension	
	Working in a Team	
	Influence, Persuasion, & Negotiation	
	Building a Social Network	
Technical Leadership	Building and Orchestrating a Diverse Team	• A few individuals indicated that training for decision-making or management skills were available through their organizations.

Proficiency Area	Category	Education & Training
	Balanced Decision Making and Risk Taking	Where available, individuals indicated that this provided some useful principles and awareness, but that this information had to be put into practice to be internalize and made more useful. • Some senior systems engineers have pursued a master's of business administration degree in order to build awareness and skill in these areas.
	Managing Stakeholders and their Needs	
	Conflict Resolution and Barrier Breaking	
	Business Management Skills	

Though education and training clearly play a role in the development of a systems engineer's proficiency, many interviewees cited roadblocks to these initiatives, particularly the lack of training or educational opportunities, either through limited offerings, unclear selection criteria, or the unwillingness of organizations to allow time for individuals to attend training.

5.3 WAY AHEAD FOR FORCES

It is clear from the discussion above that none of the forces exist in isolation and that, in fact, it is through a combination of experiences, mentoring, and education and training that a systems engineer becomes effective. Going forward, the Helix team will explore more thoroughly the interactions between these forces. Ideally, *Atlas v. 0.5* will include some recommendations on how these forces can be used synergistically to accelerate the development of systems engineers.

6. ENABLING CHARACTERISTICS

As discussed in Section 3.2, in addition to the forces of *Experiences*, *Mentoring*, and *Education & Training*, there are characteristics that can enable or inhibit a systems engineer's ability to deliver value. The following sub-sections provide additional details on the personal and organizational characteristics.

6.1 PERSONAL ENABLING CHARACTERISTICS

Descriptions of the proficiency framework, the areas and categories, provided details on various knowledge, skills, abilities, behaviors, and cognitions that systems engineers need be effective. In addition, there are certain characteristics that systems engineers possess that influence the efficiency of value delivery; they also influence the efficiency of the forces that impact the effectiveness of the systems engineer. For example, two individuals with similar educational backgrounds and experiences undergoing the same training program may accrue different levels of benefits.

In some Helix interviews, participants offered their thoughts on whether some of the abilities were innate or learned. Though such discussions were informative, they are insufficient to categorically state if someone is "born with" a particular ability or if it could be developed later on in life. Helix is not reporting any such observations, but some of those abilities are considered to be personality related characteristics and traits. These characteristics influence the proficiency of the systems engineer and the impact of the forces on those proficiencies.

- **Lifelong Learning.** Systems engineers always need to stay current with recent developments in technologies, tools, and processes. In addition, they need to constantly improve their understanding of the system's domain and related disciplines. This means that systems engineers need to be students all their life, constantly learning and increasing their knowledge, irrespective of their seniority or position in the organizational. Related to the ability to learn is also the ability to teach and share knowledge with others. It is common for systems engineers to play the role of a teacher or a professor to members of his or her team.

- **Self-Awareness.** "Problems aren't necessarily a problem; not recognizing that you have a problem is a big problem", said one interviewee. Systems engineers need to be able to self-reflect and become aware of their strengths and weaknesses; what they know and don't know; and where they are right and where they are wrong. An increased awareness of oneself is only the first step; to acknowledge wrong and to correct it requires humility and modesty as well. Such self-awareness also brings clarity to a systems engineer's mind as to where they can rely on their own knowledge and where they need to seek the expertise and experience of others. Another interviewee said, "The best systems engineers I work around are people that don't think they know everything."

- **Ambition and Internal Motivation.** Systems engineers tend to be very ambitious in terms of their career goals as well as the nature of the systems they wish to engineer. Most systems engineers who participated in Helix interviews expressed a desire to eventually become a 'Chief Systems Engineer' or play another senior role, and to also work on increasingly complex projects. Such high ambitions make systems engineers very strongly motivated internally. Even in challenging situations, they are able to generate energy and motivation within themselves, without relying on an external source. They tend to approach any problem with a 'can-do' attitude.

- **Confidence, Persistence and Focus.** Systems engineers naturally exude confidence - they need to, in order to interact with top management within the organization; with subject matter experts who are technically sound; with strong personalities who are highly opinionated; and with end customers whose requirements are to be satisfactorily met for the system to be successful. Systems engineers are not known to be shy, and they do not 'give-up' easily. They remain persistent to make things progress towards the end goal and vision for system success, which is always in their focus; and they do not get distracted easily.

- **Professionalism and Respect.** Systems engineers are very professional in their conduct, mannerisms, and behaviors; they maintain a good work ethic. Some interviewees also mentioned that systems engineers are "people with a lot of charisma", which makes them natural leaders. They treat others with respect, and acknowledge their strengths, contributions, and value they bring to the team and the organization. They are patient when they need to be, and tend to 'tolerate' difficult people.

- **Personal Interests and Experiences**. "Systems engineering is a unique combination [of] left brain – right brain working together," said one interviewee; others also indicated that there was an artistic side to systems engineering. Many interviewees elaborated much about their hobbies, interests, experiences, and life in general outside the office, and the influence of all those on their professional life as a systems engineer. In many cases, participants talked about developing the proficiencies in the SE mindset category very early on in life, and through personal interests. Among the systems engineers who participated in Helix interviews were musicians, avid readers, poets, painters, authors, puzzle-lovers, dancers, and gardeners. One puzzle enthusiast stated that she was very interested in jigsaw puzzles from a very early age, and so she approaches any problem at work with the assumption that there *is* a solution – it may not be obvious, but it exists somewhere to be discovered. Another interviewee elaborated how his life growing up on a farm taught him critical systems engineering lessons. Many confessed to be systems engineers at home as well, and that their spouses and children would attest to it.

The personal enabling characteristics discussed above, play a critical role in the development of effective systems engineers. Some interviewees who had played the role of being a mentor stated that these characteristics could also serve as indicators for identifying potentially effective systems engineers. Experiences and mentoring appear to influence the development of these characteristics. Helix will further explore how these characteristics may be instilled and developed in individuals who may be lacking in these.

6.2 ORGANIZATIONAL ENABLING CHARACTERISTICS

There are several characteristics of organizations that provide an indication of how difficult or easy it may be for a systems engineer to be effective. Some of these principles have been mentioned in the previous Helix reports, though in a slightly different context. Below is a general discussion of principles; it does not contain profiles of the organizations. As shown in Figure 6, above, the primary characteristics are:

- **The definition of "systems engineering" and "systems engineer".** As discussed in the first Helix report (Pyster et al. 2013), 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 with unclear or lacking definitions of these terms, interviewees cited several issues, particularly that individuals outside of the SE community form their own definition of what a systems engineer does based on their personal experiences with an often limited sample of systems engineers. Particularly when the title "systems engineer" is applied loosely within an organization this can cause tension, as people do not have clear expectations for what a value systems engineer should truly bring to a project. Even in organizations that have a clear definition of "systems engineer", individuals reported lack of awareness of the definition, both within and outside the systems engineering community. Interviewees tended to indicate that a clear, concise, and commonly understood definition of systems engineering would improve their ability to be effective and provide contributions on their teams. However, almost every interviewee indicated that this is not currently the case at his or her organization and, therefore, the actual efficacy of this approach remains to be seen.

- **An organization's appreciation of systems engineering and perspective on the value that systems engineers provide.** Though clearly linked with the definitions of "systems engineering" and "systems engineer" discussed above, this is the organization's formal perspective on the value that systems engineers can provide. If an organization has no value proposition or if the value proposition for systems engineers is unclear, it again raises uncertainties with individuals outside of the SE community. These individuals do not understand what their expectations from systems engineers should be or what return on investment they should expect when they allocate a portion of their budget to systems engineering activities. In the DoD environment of increasingly dwindling funding, this raises stresses in the organization and makes it more difficult for systems engineers to be effective members of their teams. Individuals stated that they often have to "fight the battle" of explaining to program managers or functional managers why they should participate – and that these actions take time away from the actual provision of value. One individual stated that the value proposition for systems engineers was a negative in her organization, stating that her project manager would tell her to "go off in [her] corner and work on [her] requirements and he would let [her] know when he needed them." Clearly, fighting this sort of cultural misperception of what systems engineers can do makes it more difficult for systems engineers to be effective.

- **An organization's overarching 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 things certainly impact the ability of systems engineers to provide value to the organization. For example, a culture that values individual contributions over team contributions is a difficult environment for a systems engineer, whose value is often realized through team coordination and interaction. Structure also impacts the organization and can compound issues with an unclear or nonexistent definition of systems engineering or systems engineers. For example, in one organization there is a unique systems engineering unit where all individuals in the unit are called "systems engineers" – but this unit contains individuals who are not practicing systems engineers or who practice a very limited scope of the systems engineering skillset. Because they hold the title of "systems engineer", individuals outside the systems engineering unit have a skewed impression of the value that a systems engineer can bring. The way systems engineers are deployed to projects – whether they become part of the reporting structure or are simply "matrixed out" – can also impact performance. Systems engineers in the matrix structure consistently report that because they have no authority – and no input on things like performance reviews – they must lead through influence only, which can be especially difficult in a climate that is already unsure of the real value of systems engineering. Organizations that do state a value proposition for systems engineers tend to make systems engineering training more available and facilitate outreach with other disciplines.

- **The organization's process for rewarding and recognizing systems engineering.** Every interviewee who discussed performance evaluation described a very common and generic annual performance evaluation system. Most stated that there were no specific outcomes or objectives in their evaluations related to the value that they provide as systems engineers. In fact, some organizations' reward systems are so focused on individual contribution that they are skewed against systems engineers – or worse, encourage systems engineers to focus on their individual performance instead of taking a holistic systems view of the team. No organization was described as having a consistent means of evaluating or rewarding systems engineering practice. However, some individuals stated that their individual managers or supervisors had generated specific systems engineering related goals that were tied to their performance evaluations and that it was helpful and encouraging to be rewarded for this performance. These measures, however, were still difficult as they often included items such as cost and schedule, which are impacted by but not under the control of the systems engineer. Metrics related to teamwork and team performance were described as desirable but difficult to implement.

- **The potential for career growth in the field of systems engineering.** Interviewees consistently stated that 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. Several interviewees also stated that the more senior systems engineering positions were filled and would not open until someone retires – meaning that individuals considering entering systems engineering know that they have little opportunity for upward mobility and may be less inclined to pursue the discipline. Only one of the organizations in the current sample had a clearly mapped career path that seemed consistently understood by the interviewees. Another organization in the sample is actively working on creating career paths so that junior systems engineers have some understanding of the overall expectations and growth opportunities within the organization.

Most of the aspects discussed above reflect a negative – something that is an inhibitor of systems engineers' effectiveness. This is due to the nature of the current data set, where impediments were consistently discussed but positives were less frequently mentioned. It seems reasonable to believe that the inverse of impediments would yield a positive return – e.g. if lack of a clear definition of "systems engineer" impedes effectiveness, then development of a clear definition would improve effectiveness – but this can not be confirmed until organizations with these characteristics are included in the sample.

Clearly, each of these aspects of an organization are linked and changes in one area would change the other characteristics of the organization. Going forward, the Helix team will work on developing a clearer understanding of the interactions between these variables.

Finally, the Helix team has data on these characteristics for each participating organization. Going forward, the team will apply this lens in data analysis to see if any patterns in the data are explicitly linked to organizational characteristics.

7. AN UPDATE OF CAREER PATH ANALYSIS

Workforce planning and improvement activities must start by identifying the subject “workforce”. As discussed in the first Helix report published in December 2013, there is no consistent definition of “systems engineer” across the community; the term is used inconsistently in titles across organizations, and in some cases even within a single organization. (Pyster et al. 2013) The analysis presented in this section seeks to answer the question of “who” systems engineers are by examining the variety of experiences that systems engineers have had, looking at career paths of junior, mid-level, and senior systems engineers. This is an update of the information provided in the second Helix report (Pyster et al. 2014) and includes roughly double the number of individuals, with a total of 69 systems engineers being included. At this time, most of the data analyzed (about 80%) for this section of the report was from defense industrial base employees, not DoD employees. This analysis will continue to be updated in Helix Phase 5.

7.1 CAREER PATH ANALYSIS METHODOLOGY

This initial analysis includes the information for 69 of the 165 systems engineers in the Helix example. The general approach for this analysis included:

- Review of resumes submitted by each individual when available; and
- Review of first interview transcripts and notes to develop a preliminary profile.
- Review of preliminary results during follow up interviews to clarify analysis. Roughly a third of the individuals in the career analysis sample have participated in follow up interviews to date.

Career path analysis examines several aspects of an individual’s career, including:

- The different lifecycle stages experienced by each individual;
- The variety of programs worked on by each individual, including program size and type;
- The variety of programs worked on by each individual, including program type and application domain;
- The number and type of organizations worked in by each individual;
- The role(s) played by each individual; and
- Identification of individuals as junior, midlevel, or senior based on this analysis and individuals’ self assessments of their seniority.

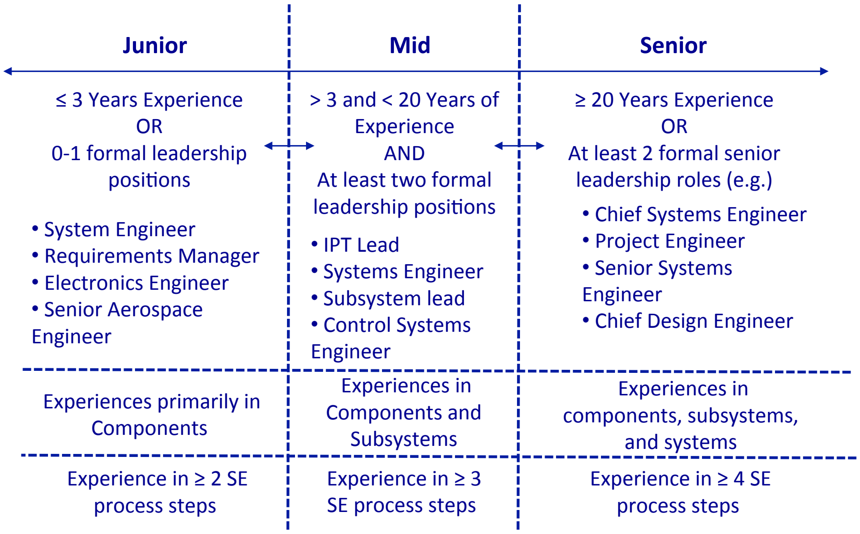
These aspects are reported in aggregate below and, where possible, with differences in gender or seniority. It is important to caveat these results with the fact that most of the data has come from resumes and the initial interviews, which did not consistently provide this level of detail for all interviewees. For example, some individuals may have worked on more types of programs than they reported in their resumes or revealed in their initial interviews. The Helix team will continue to gather additional data and validate or revise its initial findings via follow up interviews.

Note that because there are only four mid-level systems engineers in the current sample, they may be characterized in the findings, but are not always included in the figures, particularly where this would lead the reader to emphasize the information, which is inappropriate with such a small sample size.

7.2 SENIORITY LEVELS OF SYSTEMS ENGINEERS

It is useful to distinguish Helix findings based on the seniority levels of the participants, providing insight into how the findings vary between systems engineers new to the field and to those who have been practicing for many years. Placing people in levels is more difficult in systems engineering than in traditional engineering fields due to inconsistencies in titles, roles, and in identifying when someone becomes a systems engineer. For example, someone who comes into an organization with a bachelor's degree in electrical engineering usually has a clear identity as an electrical engineer. That person maintains his 'electrical engineer' identify for years, perhaps for his entire career, even as he takes on increasingly more complex technical challenges and projects. On the other hand, that same electrical engineer may begin performing systems engineering related tasks early in his career while still being identified as an electrical engineer. There is no consistent point, either across the community or even within a single organization, at which the organization recognizes that systems engineering tasks have come to dominate that individual's role and that he or she is now a 'systems engineer'.

The Helix team has developed criteria for determining the seniority of systems engineers that would allow for diversity across and within organizations. An initial version of this was provided in the second report (Pyster et al. 2014) but is refined in Figure 7.



Junior	Mid	Senior
≤ 3 Years Experience OR 0-1 formal leadership positions	> 3 and < 20 Years of Experience AND At least two formal leadership positions	≥ 20 Years Experience OR At least 2 formal senior leadership roles (e.g.)
• System Engineer • Requirements Manager • Electronics Engineer • Senior Aerospace Engineer	• IPT Lead • Systems Engineer • Subsystem lead • Control Systems Engineer	• Chief Systems Engineer • Project Engineer • Senior Systems Engineer • Chief Design Engineer
Experiences primarily in Components	Experiences in Components and Subsystems	Experiences in components, subsystems, and systems
Experience in ≥ 2 SE process steps	Experience in ≥ 3 SE process steps	Experience in ≥ 4 SE process steps

Figure 7. Preliminary criteria for identifying individuals by seniority.

The breaks between categories are heuristic, not principled, but are also largely consistent with how the interviewees tended to view themselves.

- A systems engineer with less than 3 years of experience is considered junior.
- A systems engineer who has held no more than a single leadership role, regardless of years of experience, is considered junior.[3] This aligns with data from the follow up interviews, where most participants consider junior systems engineers to be largely individual contributors, who have not yet had any formal leadership positions as systems engineers. Some junior systems engineers have held a single leadership role, generally with regards to a specific component or on a small project, such as an IR&D.

---

[3] The term "formal leadership role" is important here. Because of the nature of systems engineering, effective systems engineers at all levels must exhibit some leadership behaviors such as influencing or negotiating. The distinction here is that there must be some positional authority assigned by the organization. The Helix team will explore the differences between leadership behaviors and leadership positions going forward.

- A systems engineer with over 20 years of systems engineering experience is automatically considered senior.
- Within each organization there are certain roles, such as "program engineer", "project engineer", "chief systems engineer", and "senior systems engineer" that are only played by the most experienced systems engineers. Therefore, an individual who has played one of these roles in one of these organizations is considered senior.
- By definition, mid-level systems engineers are those who are neither junior nor senior, having some characteristics of both:
    - Mid-level systems engineers often act as both individual contributors and leaders of small teams. A mid-level systems engineer must have held at least one small-scale formal leadership role in their professional experiences.
    - Mid-level systems engineers have more than three years of experience and less than 20. But it is also important to understand what these systems engineers have done in order to classify them. For example, an individual who has five years of experience but has never had a leadership role would still be considered a junior systems engineer.

As a junior systems engineer gains proficiency, he or she will eventually become mid-level – but before reaching mid-level, he or she will improve proficiency through experiences, mentoring, or education and training. The same is true for mid-level systems engineers working towards becoming senior systems engineers. But the transitions between junior and mid-level and mid-level and senior are not clear-cut. In fact, interviewees could not define these transitions and instead focused on the ends of the spectrum – describing very junior and very senior systems engineers.

The Helix team will continue to refine the criteria used to categorize individuals by seniority. As an additional validation of the categories, the Helix team asked those who participated in follow up interviews to self-identify as junior, mid-level, or senior and to explain their rationale. The majority of self-identifications matched the Helix team's assessments. In some cases, however, an individual self-identified with a different seniority level. When the rationale for this decision provided new data that was not part of the initial Helix assessment and this information provided insight with respect to the above criteria, the Helix team updated the assessment. When the rationale did not provide new data about the individual but instead was related to organization-specific context, the Helix team retained the Helix assessment. With respect to individuals who identified themselves as mid-level, they could not clearly define what mid-level meant – when asked their rationale, the answer was consistently, "I am too senior to be a junior systems engineer, but I am not playing a senior role and probably won't for a few years."

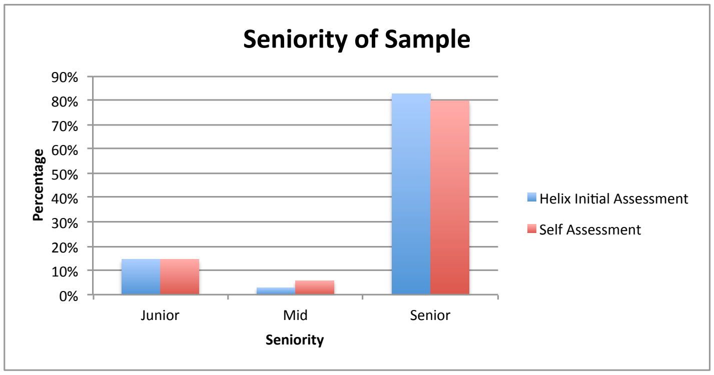


Figure 8. Overview of the sample by seniority.

Figure 8 shows that almost 80% of participants to date have been senior-level systems engineers; mid-level systems engineers are the most under-represented group in the current sample. It is unclear if this reflects the seniority of the wider population or if this is an artifact of how organizations selected interviewees. This profile does overlap the profile of the DoD systems engineering workforce by age, as seen in (Welby 2010). The Helix sample is most lacking in mid-level systems engineers; while the team will specifically seek to interview more mid-level systems engineers in future interviews, it is possible that this could be a reflection of the population.

7.3 CAREER PATHS FOR SENIOR SYSTEMS ENGINEERS

One of the goals of Helix is to understand how senior systems engineers have progressed through their careers. The team can then compare this historical perspective to the current career paths of junior and mid-level systems engineers to help determine areas of differences in early experiences between the current generation and the future generations of senior systems engineers.

Because senior systems engineers make up roughly 80% of the current sample, inferences about the larger population are more appropriate. Figure 9 provides an overview of the most typical career profile for current senior systems engineers.

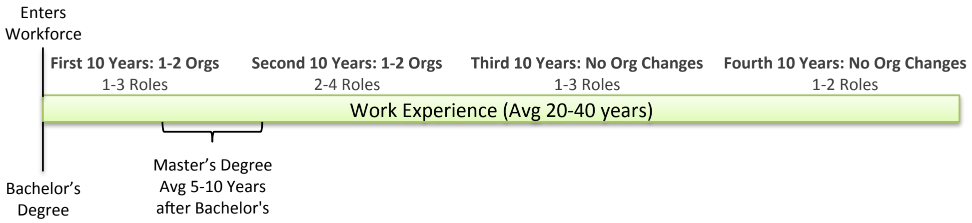


Figure 9. Typical career progression for a mid-level systems engineer.

The educational background of senior systems engineers is fairly consistent: 100% of senior systems engineers have a bachelor's degree and 69% of senior systems engineers have at least one master's degree. Figure 10 below shows the breakdown by field of study for both bachelor's and master's degrees. There are only 6 senior systems engineers with PhDs in the current sample.

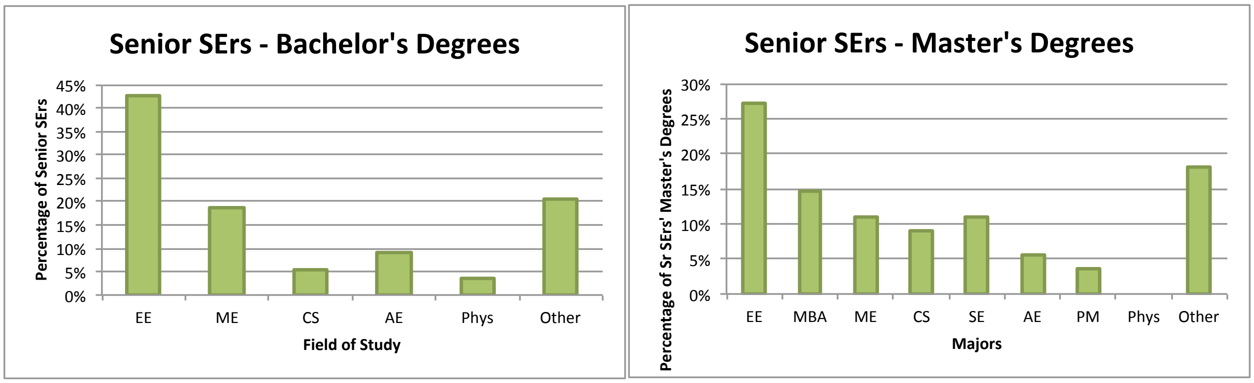


Figure 10. Overview of the common fields of study for senior systems engineers' bachelors degrees (a) and master's degrees (b).

Another aspect of characterizing the senior systems engineering population is understanding their total years of experience versus their years of experience as a practicing systems engineer. Figure 11, below, provides insight on total years of experience. However, when asked about their first systems engineering roles, many of the senior systems engineers stated that when they learned what systems engineering was, they discovered they had already been practicing it for many years. The Helix team is going through the data to identify, based on descriptions, first systems engineering roles. This analysis will be included in the next report.

Figure 11. Total years of experience for senior systems engineers in sample.

The majority of senior systems engineers have between 20 and 40 years of experience, with very few (about 3%) with 17-19 years of experience and about 17% with greater than 40 years of experience.

Most senior systems engineers started in a specialty engineering field (usually electrical or mechanical) and worked for 5-10 years before pursuing a master's degree, usually in a specialty field. For senior systems engineers with more than one master's degree, the second degree was often a master's of business administration (MBA); in follow up interviews, most interviewees with MBAs stated that they chose to get an MBA because they believed their technical skills were adequate, but they wanted to understand the business cases and business context of the systems they worked on more thoroughly.

In terms of organizations, senior systems engineers generally changed organizations 1-3 times in the first 20 years of their careers, after which they typically stayed at a single organization. Around 17% of the sample has stayed with one organization their entire career. Within each organization, senior systems engineers performed an average of 1-3 different roles. Roles were counted either as performing different roles (e.g. system analyst and requirements engineer) or performing the same role on two different systems. Roles were normalized using the descriptions developed in the second Helix report (Pyster et al. 2014), with the results displayed in Figure 12.

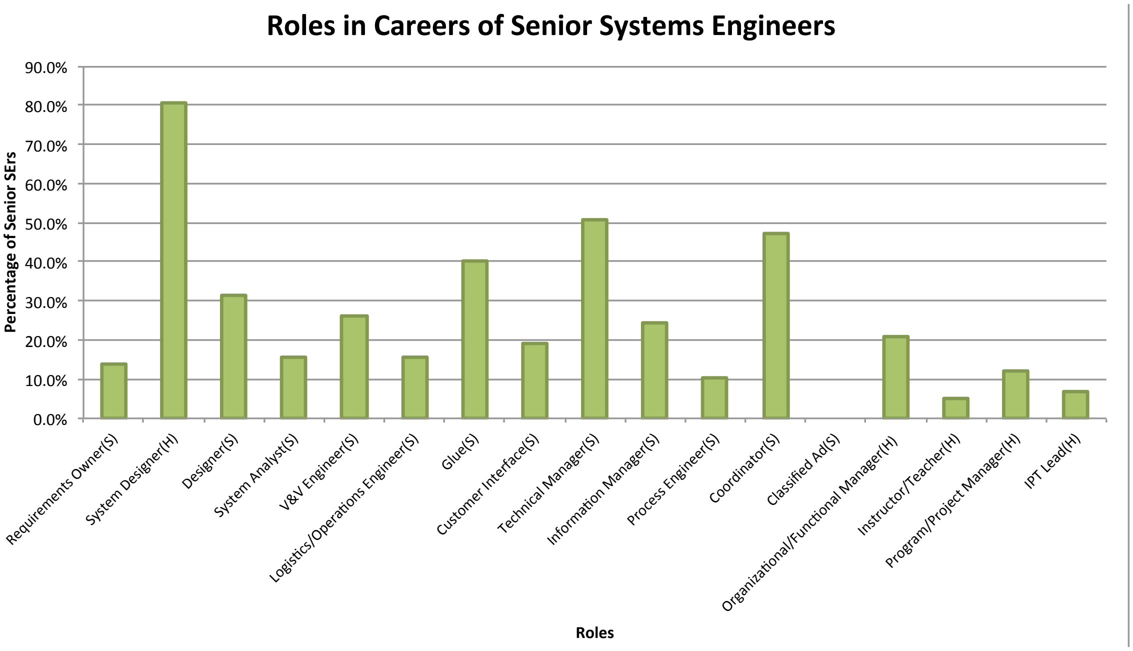


Figure 12. Roles played in the careers of senior systems engineers.

7.3.1 CAREER PATH FOR CHIEF SYSTEMS ENGINEERS

The team has scoped this analysis to a specific role that is played by senior systems engineers: a chief systems engineer. The term “chief systems engineer” is not used in all organizations participating in Helix, but is common enough to be applied. In this context:

- A **chief systems engineer (CSE)** is responsible for the technical aspects of a program, acting across the program lifecycle and providing coordination and, when necessary, negotiation across a number of engineering specialties to develop a system solution and an appropriate process to realize that solution. The CSE often has direct interaction with the customer as well as oversight of other systems engineers and/or engineering personnel within a program.

The Helix team has identified ten individuals in the current sample that are currently playing the role of CSE. Nine of these are in industry and one is in government. Because of the nature of government acquisition efforts – where government systems engineers often review work done by contractors – it is not surprising that few of the government participants to date have played the role of CSE.

Figure 13 provides an overview of the most common career path for the CSE in the sample. As the Helix team adds data to this analysis, this view of CSEs’ careers will become richer and will hopefully include more examples from government.

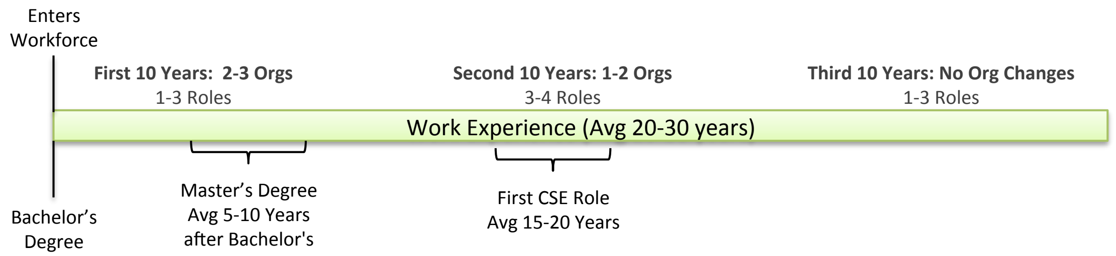


Figure 13. Overview of the common career path for chief systems engineers.

The CSEs in the sample have from 20-30 years of experience in the workforce. Most of them entered the workforce after graduating with a bachelor's degree; three individuals entered the workforce while completing their degrees. Most CSEs then stayed in the same organization for several years, with over half of them earning a master's degree 5-10 years into their careers. About 40% of the chief systems engineers worked in only one organization during their entire careers. The remaining 60% changed organizations primarily in the 10-15 year range, and then stabilized in their current organization between 15 and 20 years into their careers. Additional observations about careers for CSEs include:

- All of the CSEs in the current sample have experience across 4 stages of the lifecycle. (Details on and definitions of the lifecycles outlined in Section 4.5, below.)
    - About half gained their first experiences in *System Definition*, which includes traditional development activities.
    - Two had their first experiences in *Concept Definition*; one in *Deployment and Use* (a member of the military who maintained systems); and one in *System Realization*.
    - All but one chief systems engineer has experience in *Concept Definition*.
    - All chief systems engineers have experience in *System Definition* and *System Realization*.
    - Half of the chief systems engineers have experience in *Systems Deployment and Use*; the other half in *Systems Engineering Management*.
- The roles played throughout the careers of CSEs are outlined in Figure 14.

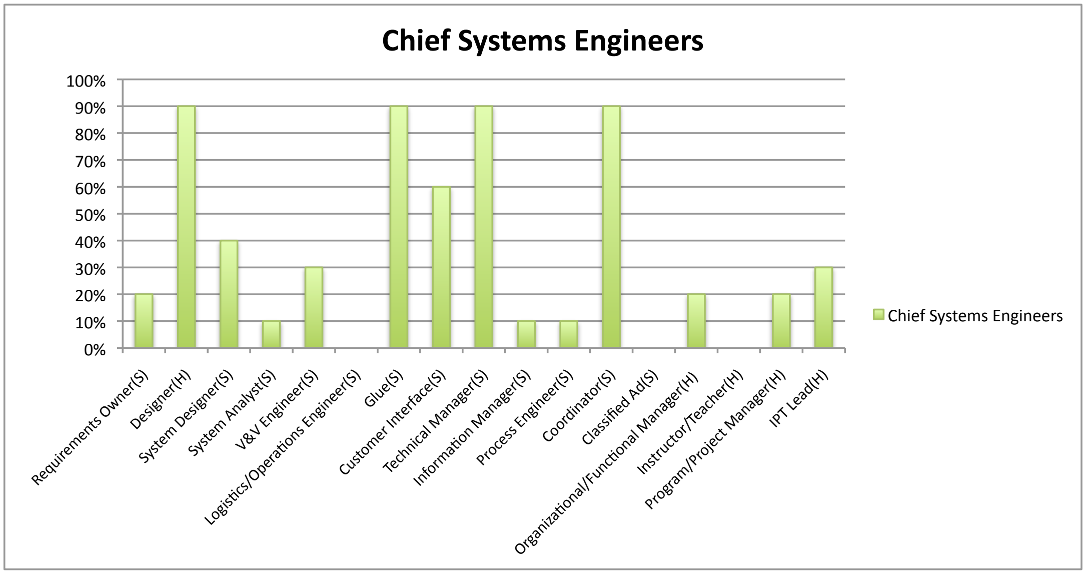


Figure 14. Roles played by chief systems engineers over their careers.

  - Though updated to reflect more of the sample, the overall characterization of the data aligns with the description in the second Helix report, so is not reproduced here. (Pyster et al. 2014)

- Regarding the education of these CSEs:
  - All have bachelor of science degrees; one individual also has a bachelor of arts degree
    - Eight of these degrees are in electrical engineering; one is in mechanical engineering, and one is unspecified. Electrical engineering is a core capability for a few of the organizations that have participated in Helix to date; therefore the focus on electrical engineering in education is not surprising and may not be representative of the wider population of chief systems engineers.
  - 60% have master of science degrees; this is slightly below the percentage of senior systems engineers with master's degrees, indicating that today's chief systems engineers may have been slightly more likely to focus on experiences over education.
  - 20% have master of business administration (MBA) degrees; this is 50% higher than the average for the total senior systems engineers in the sample.

7.4 EXPERIENCES ACROSS ORGANIZATIONS

One aspect of a career path is the number of different organizations an individual works for over the span of his career. According to interview data, working for different organizations exposes individuals to a variety of factors including multiple cultures, missions, types of systems, application domains, business processes, and systems processes. At present, the Helix team can provide specific insights into two aspects of organizational experiences: 1) the number of different organizations at which an individual has experience, and 2) whether these organizations have been in government, industry, or a mixture of the two.

7.4.1 EXPERIENCE BY NUMBER OF ORGANIZATIONS

Though counting the number of organizations seems straightforward, there were a few challenges to this analysis:

- For industry experiences, many of the companies within the defense industrial base have merged and split over time. The challenge was to determine whether these organizations should be considered as one or several. Based on guidance from senior systems engineers as to why working for multiple companies may be beneficial (explained below), the team asked about these transitions in the follow up interviews. T*he Helix team has determined that there are sufficient differences in organizational culture, process, and/or scope to consider these to be multiple organizations for this analysis.*
- For government organizations a similar question arose. This challenge primarily arose from individuals staying within a high-level umbrella organization, but moving to several different components of that organization. For example, an individual working within a program executive office (PEO), but moving to provide support to the individual program offices within the PEO. As with the industrial counterpart, the team reviewed interview data about the differences as these varying levels. *The Helix team determined that, in general, culture or scope varied greatly, and these instances are considered to be different organizations for the purposes of this analysis.*

Using the assessment criteria explained above, Figure 15 shows the profile of the number of organizations in which an individual has worked by seniority.

Figure 15.The number of organizations worked in by each individual in the sample.

Of the 41 senior systems engineers in the current analysis, around 70% have worked in three or fewer organizations during their careers, while around 30% have worked in four or more. As there are only four mid-level systems engineers in the current sample, no trends can be identified. A third of the junior systems engineers have already had experiences at three or four organizations. Several interviewees have anecdotally noted that they believe that junior members of the workforce in general are much more likely to move to new organizations in search of different opportunities. This may account for higher diversity of organizations earlier in the careers of junior systems engineers.

The strong majority of the sample in the current analysis is male; the sample size of females is too small to see any clear patterns at this time by gender about movement through organizations.

7.4.2 EXPERIENCE BY TYPES OF ORGANIZATIONS

The Helix team is interested in the types of organizations individuals have experience at in addition to number. Ultimately, the team would like to include size, scope, mission, etc. of organizations in this analysis. Due to the limited number of interviewees and participating organizations in the current sample, in order to maintain the anonymity of the participating organizations, at this time it is only possible to provide insight into whether individuals have experiences in government, industry, or both. Figure 16 provides an overview by seniority.

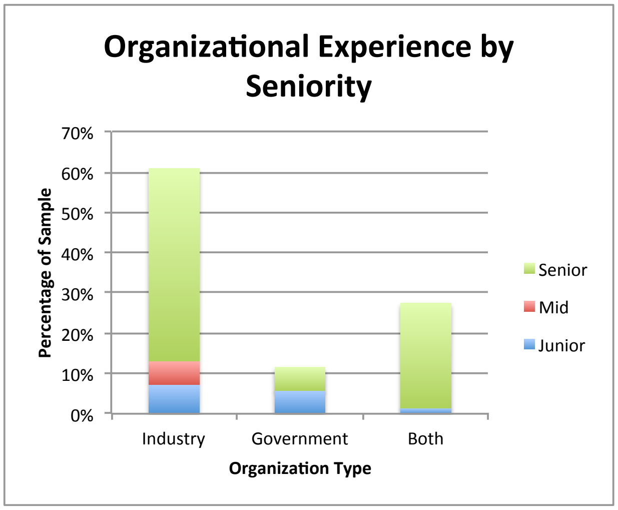


Figure 16. Percentages of the sample with experiences in government, industry, or both.

In the current sample, around 20% of individuals have only government experience, 50% have only industry experience, and around 30% have experience in both industry and government organizations.

7.5 EXPERIENCES OVER THE SYSTEMS ENGINEERING LIFECYCLE

In both the initial and follow up interviews conducted to date, individuals have stressed the importance of experiencing different aspects of the systems lifecycle in the maturation of systems engineers. To that end, the Helix team has examined the backgrounds of individuals in the current sample to determine how many areas of the lifecycle they have experienced, which lifecycle stages are most common, and in what order they were exposed to different stages of the lifecycle. Because each organization has its own processes, many of which are proprietary, the Helix team has translated lifecycle information into common terminology to protect the identities of participants and organizations and provide a mechanism for comparison across organizations. The Helix team has used the *Guide to the Systems Engineering Body of Knowledge (SEBoK)* definitions of lifecycle process activities, as outlined in Section 4.4.1 above. (BKCASE Editorial Board 2014)

7.5.1 TOTAL EXPOSURE TO SYSTEMS ENGINEERING PROCESS ACTIVITIES

Based on the resumes and interviews with each individual, the Helix team recorded which SE process activities an individual has experienced and in what order. The background information for the government systems engineers in the sample is very limited with regards to lifecycle experiences, so the information presented below applies primarily to the industrial participants.

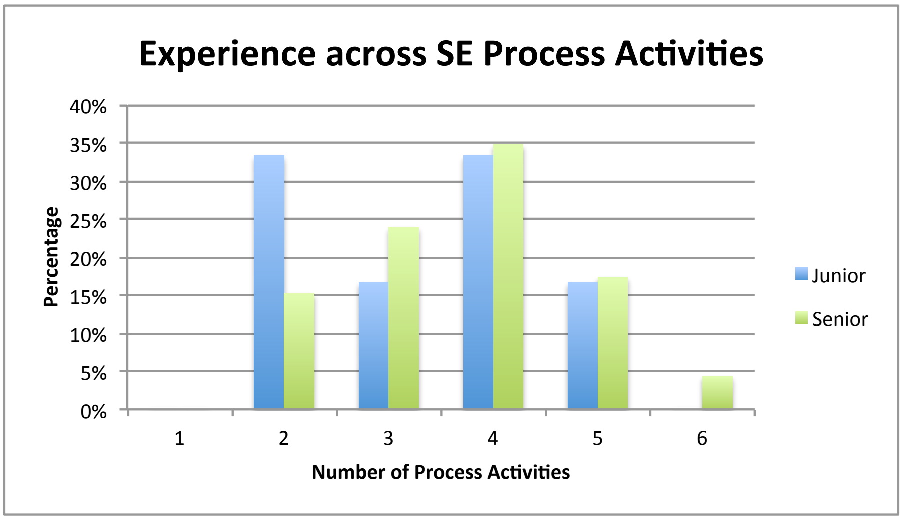


Figure 17. Breadth of experiences across the systems engineering lifecycle stages.

Figure 17, above, provides an overview the total number of lifecycle stages experienced; the percentages are of the individuals for *each seniority*, not of the whole sample. This gives an idea of the total breadth of the lifecycle an individual has experienced. For example, a "1" would mean that the individual has seen only one aspect of the lifecycle; a "6" would mean that the individual has experience in all five lifecycle phases plus systems engineering management. Most government resumes did not include this level of detail. To the extent possible, the Helix team collected data on lifecycle experiences in follow up interviews with these government systems engineers. Individuals for whom this data could not be collected were removed from the sample.

The following are findings based on this analysis:

- None of the systems engineers in the sample have experience in only one aspect of the lifecycle. It was stated repeatedly in both initial and follow up interviews that it is critical that systems engineers understand multiple stages of the lifecycle. This data seems to confirm the general belief that an individual who has experienced only one stage of the lifecycle is likely not ready to be a systems engineer.
- Less than 5% of senior systems engineers in the current sample have experience in all five lifecycle stages plus systems engineering management. In the current organizational sample, most organizations were not heavily involved in modernization or disposal efforts, therefore, this may change as the sample of organizations becomes more diverse.
- Almost 60% of senior systems engineers have experience in four or more lifecycle activities. This aligns with the finding that senior systems engineers develop in part by exposure across the lifecycle.
- The junior systems engineers show a higher level of diversity in experiences across lifecycle stages than expected. Almost half of the junior systems engineers in the sample have already gained experience in four or five aspects of the lifecycle. Several of the junior systems engineers interviewed participated in rotational programs for systems engineers who are early in their careers. In most cases the diversity across the lifecycle is related to these rotational programs. The efficacy of these programs is unknown beyond the level of exposure provided.

7.5.2 ORDER OF EXPOSURE TO THE LIFECYCLE

Though it is interesting to understand the diversity of lifecycle stages experienced by systems engineers, it is perhaps more interesting to understand the order in which individuals experience these stages. Figure 18 provides a very brief overview of the lifecycle stages and the order in which individuals experienced that stage. For example, the category "First" shows that less than 10% of individuals had *Concept Definition* as the first lifecycle stage which they experienced in their careers; almost 60% had *System Definition*; less than 20% had *System Realization*, and so on.

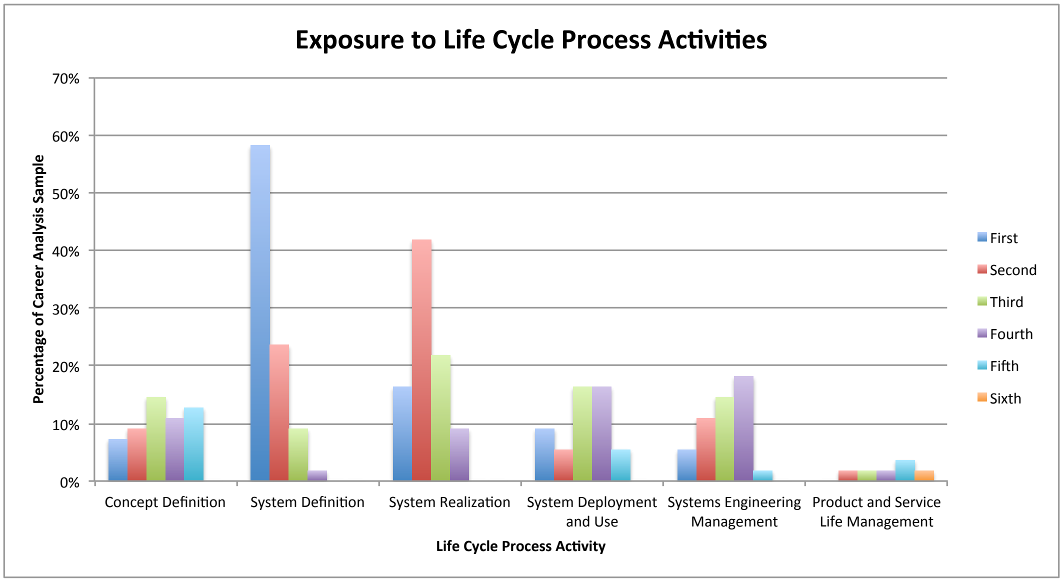


Figure 18. Order in which individuals are exposed to different stages of the systems engineering lifecycle.

A few high-level observations include:

- Over 80% of the current sample is exposed to *System Definition* as either the first or second lifecycle stage they experience. This aligns with the idea that the design of systems in a specialty engineering role is a common a precursor to systems engineering work.
- Over 60% of the sample experience *System Realization* as either the second or third stage of the lifecycle to which they are exposed. Many interviewees stated that understanding what was required to test as a system (a key activity in *System Realization*) was important for making systems engineers understand the consequences of early lifecycle decisions. Therefore, it is seen as positive that a large percentage of the sample has some experience in this area.

7.7 EXPERIENCE IN APPLICATION DOMAINS

Another aspect of an individual's career is the variety of domains in which they apply engineering or systems engineering practices and principles. The Helix team did not begin with an expected list of domains, but has catalogued domains as data has become available. Figure 19 shows the breakdown of domain experience for the current sample.

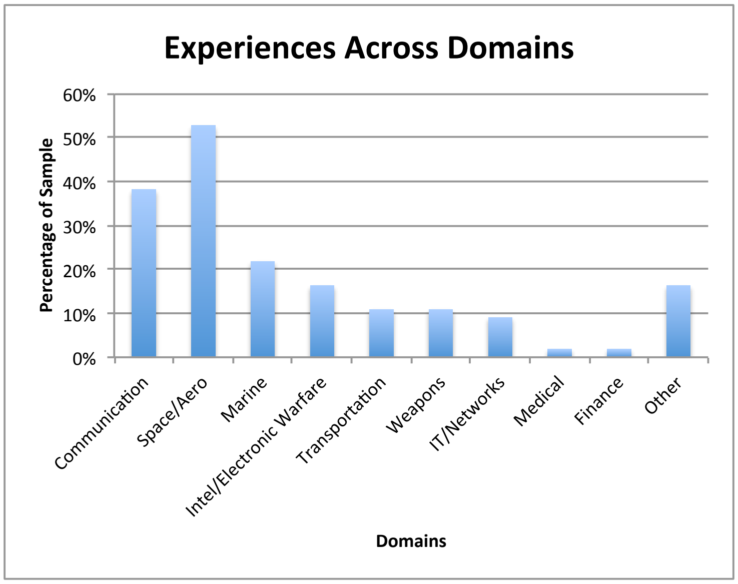


Figure 19. Breadth of domain experiences in the current sample.

The results in Figure 19 are relatively self-explanatory. In the follow up interviews, several individuals who have experience in diverse domains were asked how that has helped them as systems engineers. Some stated that it is helpful to identify common principles that can be applied across all domains. Others stated that they have seen no differences in their roles between domains, so it does not appear to provide any specific advantage. The Helix team will further explore this in subsequent interviews.

7.8 FUTURE DIRECTIONS FOR CAREER PATH ANALYSIS

Going forward, the Helix team will collect and analyze additional data and work towards building a comparable, standardized career profile for each systems engineer in the sample.

7.8.1 ADDITIONAL DATA COLLECTION AND ANALYSIS

As mentioned above, the Career Path Analysis is still evolving and the team is working to close gaps in the current data. For example, the Helix team would like to understand a wide scope of experiences related to systems and programs, including:

- Program sizes (dollar value, number of individuals on team, etc.);
- System type (product, platform, service, enterprise); and
- System level (component, subsystem, system, system of systems).

The hope is that, an understanding of the range of programs and systems that individuals have worked on, may provide a useful basis for comparison when considering current capabilities as well as providing direction for career planning. The types of data required to complete this analysis is sparse in the currently data set. In fact, with respect to system type and system level, only a few observations can be made:

- All individuals in the sample have worked on at least one product;
- All individuals in the sample have worked on at least a subsystem and system.

The Helix team is conducting follow up interviews and asking interviewees to review their career profiles and provide clarifications about their experiences with respect to these aspects of their careers.

Program size is another area for exploration. The current range of values is diverse: for the few individuals who have provided information on program size, teams range from 2-3 to over 1,000 and values from a few $100K to $10B. The Helix team hopes to provide insight into the roles of systems engineers relative to program size going forward. Some individuals have expressed concern about sharing the size of programs (either by size of the team or monetary value) that they have worked on, which may limit the ability of the team to collect this data.

7.8.2 END GOAL: COMPARABLE CAREER PROFILES

The Helix team plans to continue this analysis in the following ways:

- Complete this initial analysis for all individuals who have participated in the project (this analysis has been done for over 40% of the total sample to date);
- Work with individuals to validate career profiles in follow up interviews or through other means, including collecting additional information to fill gaps (e.g. program sizes, system types, system levels, etc.); and
- Perform this analysis with all new interviewees added to the project in Phase 4.

Once this analysis is completed, the Helix team will work on building master career profiles linked to current roles for the individuals in the study. If successful, this will allow comparisons between individuals in different roles or at different levels, and may be used to help determine potential next steps in a career path.

Ideally, the Helix team will be able to develop indicators that translate the experiences an individual has had to their probably proficiency level, as outlined in the proficiency model above. Work on *Atlas v. 0.5* will include exploration of this area, though it is unlikely that it will be finalized before version 1.0 is published.

8. FUTURE DIRECTIONS

The development and possible early application of *Atlas v. 0.5* will be the key focus of the next phase of Helix. This will be a mature draft with much more detail in the description of the variables, particularly in how these variables may be assessed. The theory will be mature enough that it an individual or an organization could implement it with some teaching and coaching from the Helix team. It will also include some validation using data from outside the DoD.

Going forward, the Helix team will continue to refine the model in several ways:

- Follow up interviews will include questions that probe parts of the model not discussed in initial interviews;
- Some initial interviews will be focused exclusively around the model;
- The Helix team will work with individual systems engineers playing specific roles to determine what knowledge, skills, abilities, behaviors, and cognitions are most relevant for these roles, creating common profiles for typical systems engineering roles;
- The Helix team will coordinate with a few organizations to work through how implementation of the model would work; and
- The Helix team will hold a second detailed Workshop focused around further refinement of the model.

The team anticipates that it will be published in late 2015.

The Helix team plans to conduct a second workshop in spring or summer 2015. The objectives of this workshop will be to aid the validation of *Atlas v. 0.25* and steps towards creating *Atlas v. 0.5* and to gain insight on potential implementation approaches and challenges for organizations wishing to use the theory.

The primary mode of data collection for Helix thus far has been through face-to-face interviews at an organization, with follow-up telephone interviews. During the next phase of the project, the Helix team expects to collect data from US Government organizations outside the DoD as well as commercial organizations outside of the defense sector. In addition, a small number of interviewees may be selected outside of systems engineering, for example program managers, software engineers, or electrical engineers. It is expected that these individuals will provide validation of the findings reported here on the value of systems engineers, the organizational value propositions for systems engineers, and the overlap in systems engineering-related proficiencies between these disciplines.

Through a Memorandum of Understanding with INCOSE, the Helix team is in the process of analyzing data from over 3,000 applications to the INCOSE Certified Systems Engineering Professional program. This results of this analysis will be incorporated into *Atlas v. 0.5.*

9. REFERENCES

Adcock, R., A. Pyster, J-C. Roussel, D.D. Walden, W.D. Miller, K. Forsberg. 2012. "What is the Value of the SE Body of Knowledge for INCOSE and SE?" International Council on Systems Engineering (INCOSE) 22nd Annual International Symposium, July 9-12, 2012, Rome, Italy.

Anderson, C. 2010. "Presenting and Evaluating Qualitative Research." *American Journal of Pharmaceutical Education.* 74(8): 141.

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

Boehm, B., J. Bayuk, A. Deshmukh, R. Graybill, J.A. Lane, A. Levin, A. Madni, M. McGrath, A. Pyster, S. Tarchalski, R. Turner, and J. Wade. 2010. *Systems 2020: Strategic Initiative*. Hoboken, NJ: Systems Engineering Research Center (SERC). SERC-2010-TR-009.

BKCASE Editorial Board. 2014. The Guide to the Systems Engineering Body of Knowledge (SEBoK), v. 1.3. R.D. Adcock (EIC). Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed 14 November 2014. www.sebokwiki.org.

Creswell, J.W., C.V.L. Plano. 2011. *Designing and Conducting Mixed Methods Research*. 2nd ed. Thousand Oaks, CA: Sage.

Davidz, H.L. and D.J. Nightingale. 2007. "Enabling Systems Thinking to Accelerate the Development of Senior Systems Engineers." *Systems Engineering*. 11(1).

DoD. 2013. "SPRDE Functional Career Field: Critical Acquisition Workforce Data FY 2013-Q3 (as of June 20, 2013)." U.S. Department of Defense, Office of the Undersecretary of Defense for Acquisition, Technology, and Logistics (OUSD (AT&L)). 30 June 2013.

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

Ferris, T., D. Prun, K. Geist, A. Chia, P. Jackson, R. Freeman. 2011. "Graduate Reference Curriculum in Systems Engineering." International Council on Systems Engineering (INCOSE) 21st Annual International Symposium, June 20-23, 2011, Denver, CO, USA.

Frank, M. 2006. "Excellence in Systems Engineering: What Characterizes Successful Systems Engineers?" (paper #106). Conference on Systems Engineering Research (CSER), April 7-8, 2006, Los Angeles, CA. 1-10.

Frank, M., K. Frampton, and T.D. Carlo. 2007. "Characteristics of Successful Systems Engineers, Systems Architects and IT Architects." Asia-Pacific Software Engineering Conference (APSEC), Nagoya, Japan. 1-12.

GAO. 2008. *Defense Acquisitions: Assessments of Selected Weapon Programs*. Washington, D.C.: Government Accountability Office (GAO). GAO-08-467SP.

———. 2011. *Weapons Acquisition Reform: Actions Needed to Address Systems Engineering and Developmental Testing Challenges*. Washington, D.C.: Government Accountability Office (GAO). GAO-11-806.

———. 2012. *Weapons Acquisition Reform: Reform Act Is Helping DOD Acquisition Programs Reduce Risk, but Implementation Challenges Remain*. Washington, D.C.: Government Accountability Office (GAO). GAO-13-103.

———. 2013. *High Risk Series: An Update*. Washington, D.C.: Government Accountability Office (GAO). GAO-13-283.

Gosling, S.D., P.J. Rentfrow, and W.B, Swann Jr. 2003. “A Very Brief Measure of the Big-Five Personality Domains.” Journal of Research in Personality 37 (6): 504–28

Goulding, C. 202. *Grounded Theory: A Practical Guide for Management, Business and Market Researchers*. Thousand Oaks, CA: Sage.

INCOSE. 2010. *Systems Engineering Competencies Framework*. San Diego, CA, USA: International Council on Systems Engineering. INCOSE-TP-2010-003.

INCOSE. 2012. *Systems Engineering Handbook: A Guide for System Lifecycle Processes and Activities*, version 3.2.2. San Diego, CA, USA: International Council on Systems Engineering (INCOSE), INCOSE-TP-2003-002-03.2.2.

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

INCOSE Technical Operations. 2007. *Systems Engineering Vision 2020*. San Diego, CA, USA: International Council on Systems Engineering. INCOSE-TP-2004-004-02.

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.

Likert, R. 1932. “A Technique for the Measurement of Attitudes.” New York: The Science Press.

MacKenzie, S.B., P.M. Podsakoff, and N.P. Podsakoff. 2011. “Construct Measurement and Validation Procedures in MIS and Behavioral Research: Integrating New and Existing Techniques.” MIS Quarterly 35 (2): 293–334.

Management Mentors. 2014. “Definition of Mentoring, Benefits of Mentoring, & Other FAQs: What is Mentoring?” Online. Accessed 30 August 2014. Available at: http://www.management-mentors.com/resources/corporate-mentoring-programs-faqs/#Q1.

“Mentor.” 2014. Merriam-Webster.com. Accessed 16 April 2014. http://www.merriam-webster.com/dictionary/mentor.

Minter, R.L. and E.G. Thomas. 2000. “Employee Development through Coaching, Mentoring and Counseling: A Multidimensional Approach.” Review of Business 21(1/2): 43–47.

Murray, M. 2002. *Beyond the Myths and Magic of Mentoring: How to Facilitate an Effective Mentoring Process*. John Wiley & Sons.

Murray, M. 2006. “Innovations in Performace Improvement with Mentoring.” In *Handbook of Human Performance Technology.* San Francisco: Pfeiffer, 455–77.

PMI. 2012. *PMI Lexicon of Project Management Terms*, Version 2.0. Newton Square, PA: Project Management Institute (PMI).

Pyster, A. (ed.). 2009. *Graduate Software Engineering 2009 (GSwE2009): Curriculum Guidelines for Graduate Degree Programs in Software Engineering*. Integrated Software & Systems Engineering Curriculum Project. Hoboken, NJ, USA: Stevens Institute of Technology, September 30, 2009.

Pyster, A., D.H. Olwell, T.L.J. Ferris, N. Hutchison, S. Enck, J. Anthony, D. Henry, and A. Squires (eds.). 2012. *Graduate Reference Curriculum for Systems Engineering (GRCSE™)*. Hoboken, NJ, USA: The Trustees of the Stevens Institute of Technology. Available at: www.bkcase.org/grcse/.

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.

Rhodes, C. and S. Fletcher. 2013. “Coaching and Mentoring for Self-Efficacious Leadership in Schools.” *International Journal of Mentoring and Coaching in Education* 2(1): 47–63.

SEI. 2007. *Capability maturity model integrated (CMMI) for development, version 1.2*. Pittsburg, PA, USA: Software Engineering Institute (SEI)/Carnegie Mellon University (CMU).

Senge, P.M. 2006. *The Fifth Discipline: The Art and Practice of the Learning Organization*, 2nd ed. New York, NY, USA: Doubleday Currency.

Shea, Gordon F. 1994. *Mentoring : Helping Employees Reach Their Full Potential*. New York: AMA Membership Publications Division, American Management Association.

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

“Theory.” 2014. Merriam-Webster.com. Accessed 14 November 2014. http://www.merriam-webster.com/dictionary/mentor.

Torelli, N. 2012. Presentation as part of Pyster, A.; Torelli, N.; Madni, A.; Hanneman, L.; Highland, F. 2012. “Systems Engineering Degrees and Certificates: What is the Right Mix for Industry and Government?” *National Defense Industrial Association (NDIA) 15th Annual Systems Engineering Conference*, October 22-25, 2012, San Diego, CA.

Walumbwa, F.O., B.J. Avolio, W.L. Gardner, T.S. Wernsing, and S.J. Peterson. 2008. “Authentic Leadership: Development and Validation of a Theory-Based Measure†.” Journal of Management 34

Welby, S. 2011. “DoD Systems Engineering: Status Report.” National Defense Industrial Association (NDIA) Systems Engineering (SE) Division Meeting, Washington D.C. Metropolitan Area, USA. 1-33.

Welby, S. 2010. “DoD Systems Engineering Update.” National Defense Industrial Association (NDIA) Gulf Coast Chapter: 36th Air Armament Symposium, Fort Walton Beach, FL, USA. 1-25.

10. GLOSSARY

ACRONYMS & ABBREVIATIONS

Acronym/Abbreviation	Meaning
CSE	Chief Systems Engineer
DASD(SE)	US Deputy Assistant Secretary of Defense for Systems Engineering
DIB	Defense Industrial Base
DOD	US Department of Defense
HAP	Helix Advisory Panel
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, 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* (Pyster and Olwell 2013)
SME	Subject Matter Expert
SPRDE	Systems Planning, Research, Development, and Engineering
UARC	University-Affiliated Research Center
V&V	Verification & Validation

TERMS

category **-** specific types of knowledge, skills, abilities, behaviors, and cognitions with shared characteristics; categories are grouped together into ***proficiency areas***.

characteristic – a feature or quality belonging to a systems engineer and serving to identify systems engineers as a group; it is not about what the systems engineer knows or can do (see ***competency)***.

chief systems engineer – Individual responsible for the technical aspects of a program, who actd across the program lifecycle and provides coordination and, when necessary, negotiation across a number of engineering specialties to develop a system solution and an appropriate process to realize that solution. The CSE often has direct interaction with the customer as well as oversight of other systems engineers and/or engineering personnel within a program.

competency – Knowledge of and skill in the practices required for successful development of a system. There are 3 types of relevant competencies for Helix:

- Technical competency – competency relevant to a specific technical discipline or domain, e.g., electrical, mechanical, or civil engineering disciplines or the telecommunications, engines, or ship domains

- Systems engineering competency – competency relevant to the process, methodologies, tools, or concepts of systems engineering
- Business competency – knowledge and skill in navigating the specific workings of the organization in which one works

effectiveness – The ability to consistently deliver systems engineering values within an organizational context.

experience – Participation in or observation of activities during which an individual is afforded the ability to learn.

- Professional experience is direct observation of, participation in, or leadership of activities in a work environment.
- Academic experience includes any activities that occur within an educational setting and generally will include formal classroom activities such as participation in a degree program.
- Social experience is any life activity (outside the classroom or professional setting) that is relevant to the effectiveness of systems engineers.

Notes: The Helix team focuses on professional and academic experience, but does collect information on social experience when provided. In general, the Helix team focuses on suites of experiences rather than a single, isolated experience.

***proficiency* –** The quality or state of knowledge, skills, abilities, behaviors, and cognitions.

***proficiency area* –** Groupings of related knowledge, skills, abilities, behaviors, and/or cognitions.

***proficiency level* –** The extent to which an individual has attained certain knowledge, has the abilitiy to perform a certain skills, or has demonstrated relevant abilities, behaviors, or cognitions.

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

systems engineering – An interdisciplinary approach and means to enable the realization of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem. (INCOSE 2012)

topic – The most discrete areas of knowledge included outlined in proficiency. Within the proficiency model, ***topics*** fall within ***categories***.