SYSTEMS ENGINEERING
Research Center

Atlas 0.6: Expanding on the Theory of Effective Systems Engineers

Technical Report SERC-2016-TR-106

April 22, 2016

Research Team:

Dr. Art Pyster, Dr. Devanandham Henry, Dr. Nicole Hutchison, Megan Clifford

Stevens Institute of Technology

Work conducted under Research Topic 154

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

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

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

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

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

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

We also thank all former members of the Helix research team whose contributions have shaped our research over the years.

- The Helix Team

EXECUTIVE SUMMARY

Research conducted through the Helix project has now produced version 0.6 of *Atlas*[1]: the *Theory of Effective Systems Engineers. Atlas* 0.6 is an *incremental* step in the evolution of *Atlas* that provides:

- Additional details into the Helix methodology, including explanations of the qualitative data analysis techniques the team has employed over the life of the project, and demographics regarding the current, full dataset.
- Description of the Helix method of career path analysis and visualization, and tools that will enable individuals and organizations to visualize career paths and to perform some level of analysis on their own.
- Incremental improvements on specific elements of *Atlas* based on additional analysis of the existing data.

The insights provided in this report are *in addition* to the insights provided in *Atlas 0.5* and so, this report should be read in conjunction with that one. In particular, more information is provided on training (one aspect of the forces), personal development initiatives, and organizational development initiatives. Between this report and *Atlas 0.5,* all elements of the theory have been analyzed based on existing data. Three specific relevant findings are:

- In training, 67% of feedback on best practices centered around ensuring that individuals have opportunities to apply learning from training in practice during or immediately after training. Likewise, 17% of the problems with training described issues when training was made ineffective because it was not quickly applied in practice. This aligns with findings from the literature about the importance of the "opportunity to perform" (Burke and Hutchins 2007) for training transfer.
- Though most systems engineers expressed concerns about the loss to the workforce when very senior systems engineers retire, only about 5% of the senior systems engineers in the sample had created training courses to improve the transfer of their knowledge to others in the workforce. Most senior systems engineers did perform some kind of mentoring, but those who had created courses described this as key to disseminating knowledge more broadly.
- The factors that stood out as most critical for success of organizational development initiatives were: establishing the right initiatives; effectively spreading the word about initiatives to employees; periodically evaluating initiatives and modifying them when needed; and gaining clear commitment from leadership supporting the initiatives.

The Helix team continues to develop *Atlas* into a complete and mature theory that may be deployed independently by individuals developing their personal careers in systems engineering, and by organizations developing their systems engineering workforce. Going forward, the team will focus more explicitly on the relationships and interactions between the elements of *Atlas* and will work with organizations to use and implement the findings of *Atlas*. The findings and outcomes of these research activities will be reported in the Atlas 1.0 report to be published during late 2016.

---

[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".

1 INTRODUCTION TO *ATLAS 0.6*

1.1 BACKGROUND

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 2006; 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 or proficiencies 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.

1.2 OVERVIEW OF HELIX

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

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

During 2015, Helix continued to collect and analyze data, defined and analyzed the career paths of systems engineers, and conducted implementation trials of *Atlas*. All this resulted in the publication of *Atlas 0.5* at the end of 2015. Discussion of these topics beyond what is reported in *Atlas 0.5* is included in this report.

1.3 OVERVIEW OF REPORT STRUCTURE

This technical report is written as an addendum to *Atlas 0.5*, presenting version 0.6 of *Atlas: The Theory of Effective Systems Engineers*. ***The reader must read* Atlas 0.5 *(Pyster et al. 2015a) before reading this report to have a proper understanding of the current state of* Atlas.** Several earlier published Helix papers and technical reports are referred to throughout this report, but the reader is not required to read them in order to understand the content presented here.

Readers should note the following about the report:

- Throughout the report, the term 'Helix' is used to denote either the project or the team that performed the work in developing *Atlas*.
- All insights and observations are presented only in an anonymous, aggregated manner. Individual systems engineers and organizations that participated in the Helix project are neither named nor are they identifiable from this report.

The report is divided into the following sections:

- Section 1 - this introduction.
- Section 2 – additional detail on the Helix research approach, including insights on the qualitative data analysis approaches used by the team. This does *not* represent changes to the Helix research methodology, but simply provides more depth and clarity in certain areas.
- Section 3 - career path analysis and visualization method. This provides detailed guidance on how to assess, document, analyze, and visualize career paths. In addition, some of the tools used by the Helix team for career path self-assessments are provided in appendices. This will help individuals and organizations begin to utilize this method on their own.
- Section 4 – updates for *Atlas 0.6* based on additional analysis. Specifically, this section highlights analyses on:
    - Training (one of the three primary Forces that impact proficiency),
    - Personal Development Initiatives, and
    - Organizational Development Initiatives.
- Section 5 – planned work that will lead to the release of *Atlas 1.0* in late 2016.

2 HELIX RESEARCH APPROACH

The analysis for the overall Helix project has not changed since the publication of *Atlas* 0.5, as outlined in Pyster et al. (2015a and 2015b). The information presented here provides additional detail on the approach, specifically:

- **Updates to the Dataset** – *Atlas 0.5* was based on a DoD and DIB population. In December 2015, the Helix team published an expansion of the theory looking outside of the defense community and at the peers of systems engineers, namely project managers and classic engineers (electrical, mechanical, software, etc.). For *Atlas 0.6* and beyond, the analysis will reflect the complete dataset, including representation from systems engineers' peers and from outside the defense community. This will make the findings of *Atlas* more robust and more broadly applicable.
- **Qualitative Data Analysis** – again, this is not a change from the approaches used but provides more detail on how Helix conducts qualitative analysis, including the specific methods and tooling used.

2.1 OVERARCHING RESEARCH APPROACH

Helix is primarily a qualitative study, with the primary means of data collection being interviews with systems engineers. From 2012-2013, the Helix team focused on a mixed-methods approach (Creswell and Plano 2011), combining the development of basic research questions with a grounded theory approach. Grounded theory was developed in the social sciences as a method for developing theory that is grounded in data that is systematically gathered and analyzed. (Goulding 2002) This approach allows the data itself to drive points of further inquiry, guide categorization, etc.; rather than starting analysis with an existing framework, all of the data is reviewed holistically and any potential areas of interest are coded. Over time patterns emerge and these guide further data collection and analysis. The development of driving research questions makes the Helix project mixed method as opposed to pure grounded theory.

When performing initial data coding, the Helix team coded all data, not making suppositions about which data would prove "important". The team also compared data collected against the existing literature where possible. For example, as systems engineers defined the activities that they perform, the team collected and organized the raw data but also compared it to the "Twelve Roles of Systems Engineers" defined in (Sheard 1996).

This approach has not changed since *Atlas 0.5*.

2.2 OVERARCHING HELIX DATASET

The total Helix dataset now includes 287 individuals from 20 organizations. These individuals are a mix of systems engineers and those who work with them – executive leaders who used to be systems engineers, organizational managers who lead systems engineers, classic engineers (e.g. electrical, mechanical, software engineers), and program/project managers (PMs). Because the focus of Helix is on what makes systems engineers effective, the vast majority of the sample (91%) was made up of systems engineers and those who used to be systems engineers – executive leaders and organizational managers of systems engineers; 9% of the sample was made up of their peers – PMs, classic engineers, etc. Over three-quarters of the interview subjects were male (78%), while 22% were female.

Among systems engineers, the Helix team developed a classification system for identifying the seniority of a systems engineer. This was validated by self-assessments and insights from the systems engineers' organizations. The seniority profile of the systems engineers in the sample can be seen in Figure 1a. Among systems engineer's peers interviewed, most (52%) were project managers; the remaining 48% were a mix of other engineers, with a few representatives from logistics management, as shown in Figure 1b.

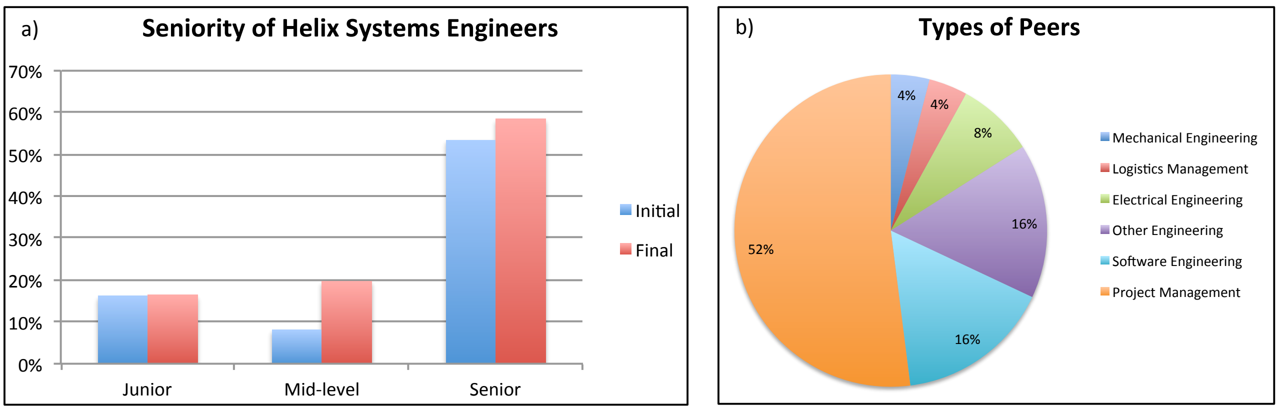


Figure 1. Demographics of the Helix sample.

Each of the 20 organizations that participated in Helix was located within the US. Some corporations were not US-based, but the team engaged with systems engineers at a US location. Because the initial motivation for Helix was to understand the effectiveness of systems engineers in the defense industry, many of the organizations are from the DoD or DIB. However, other sectors were represented as well, including healthcare, transportation, telecommunications, and information technology; and both commercial and government organizations participated in Helix. The representation of individuals from each type of organization is shown in Figure 2, below.

Participating organizations were identified based on their reputation for systems engineering or their participation in the systems engineering community. For example, several of the organizations are members of the INCOSE Corporate Advisory Board (CAB). Organizations also had to be willing to make their systems engineers available to the Helix team for at least one on-site visit; coordinate the logistics for the visit; and pay for the time their employees spent participating in Helix. In addition, organizations had to agree to abide by the rules of the Helix project, including protecting the identity of all participants and understanding that they would not receive raw data for their organizations – they would only get a summary of findings. This was the model for used for 13 of the organizations. When the Helix team expanded outside of the DoD, it was not always possible to get organizations to agree to a full site visit. For these organizations, a smaller number of individuals were interviewed over the phone, instead of in person, though still with the permission of the organization. In these interviews, fewer questions were focused on the individuals themselves; instead, the majority of the interview focused on the organization and the workforce overall.

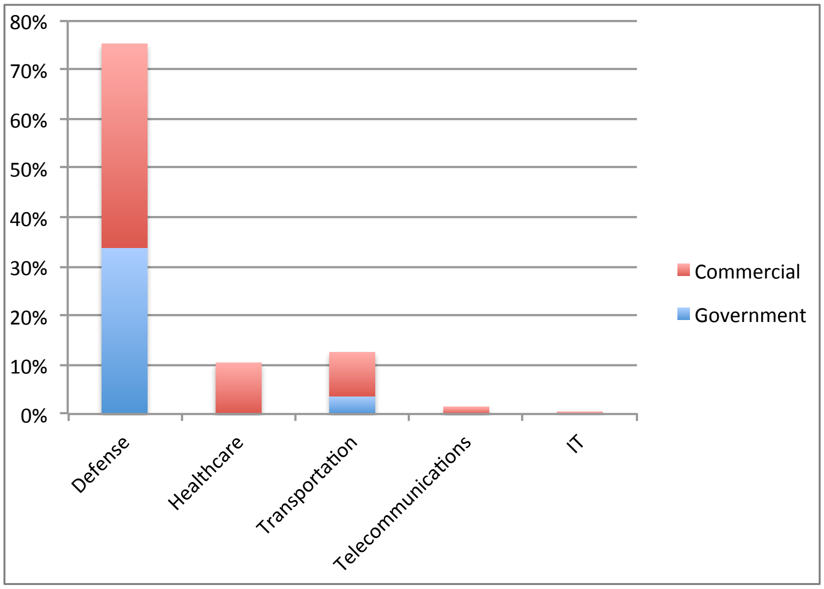


Figure 2. Individuals by organization type

Individuals were selected by the organizations based on criteria provided by Helix; i.e., the sample should be a reasonable representation of the population as a whole with respect to: gender, seniority, diversity of systems worked on, and diversity of roles played. Moreover, every organization was asked to provide only systems engineers the organization viewed as effective. The rationale for this was that individuals who were ineffective as systems engineers would likely not provide useful insights or would provide insights that would result in inappropriate recommendations. The Helix team also asked participants to assess their own effectiveness to verify the organization's perspective. After the interviews were completed, the organization was asked to re-affirm that the individuals who participated were seen as effective. All organizations stated that their interviewees ranged from "average" or "typical" effectiveness to the most highly effective systems engineers in their organizations. This was done collectively, not for each individual systems engineer, because organizations generally were not comfortable providing data to that level of detail.

Finally, each individual had to voluntarily agree to participate in the study, signing a consent form.

2.2.1 Interpretation and Generalization Using the Dataset

Helix is careful when using the data to understand whether and how findings and conclusions about the dataset can be generalized to the wider population of systems engineers. The team recognizes that there are some limitations based on the sample. Though it is relatively large at nearly 300 individuals, there is no clear estimate of how many systems engineers are working in the US, let alone the world; this makes it difficult to understand how statistically representative the sample may be. Likewise, all interviews conducted to date have been in the US; though a few individuals provided insight into, for example, education outside the US, and some organizations had units outside the US, current findings reflect a US context. Likewise, though Helix has expanded beyond its initial defense roots to include healthcare, transportation, telecommunications, and other industries, Figure 3 clearly shows that the majority of individuals participating in the sample are from the defense industry. Given these limitations, the Helix team is careful not to over-interpret the data.

However, even with the limitations of the sample, the team believes the overall size of the sample and the diversity in terms of industries, organization types, and seniority makes any findings and conclusions drawn from the data extremely useful, even if they are not fully generalizable. The Helix team first published *Atlas* 0.25 in 2014; since then the coverage of industries and organizations has grown and the team has nearly doubled the size of the sample. And over this time, there have been some updates and edits, but no major issues or breaks with the theory have been discovered. This, again, builds confidence that the existing sample is sufficient to enable useful and insightful work. As the team continues work on implementation and application of *Atlas* (see Future Directions), these activities should generate greater confidence in the generalizability of the data.

2.3 QUALITATIVE ANALYSIS

The Helix research methodology discussed in the preceding section was deployed using the research process described below.

2.3.1 CODING

The interview dataset comprises nearly 6,000 pages of transcripts and summaries from 287 individuals. In order to make sense of such a large quantity of data, the Helix team uses qualitative data analysis, primarily through data coding. Coding is "a systematic way in which to condense extensive data sets into smaller analyzable units through the creation of categories and concepts derived from the data." (Lockyer 2004) Codes can be layered, and evolve over time, as explained below. When developing a theory, as in the development of *Atlas*, categories and codes are generated *after* examining the collected data, aligning with the grounded theory approach. (Bourque 2004 and Lockyer 2004) The main type of coding done by the team so far is called "open coding", the purpose of which is to break down, compare, and categorize data (Strauss and Corbin 2014).

The team has used two techniques for coding: auto coding and manual coding. "Auto coding" is only the first stage in parsing the information contained in transcripts and is not fully automated despite its name. Instead, as the team reviews and cleans each transcript, headings are added to the source documents to block out a large area of text as addressing a particular topic, such as personal characteristics, mentoring, experiences, etc. When the documents are imported into the qualitative analysis tool, NVIVO, the tool then automatically codes all text under that heading for the given subject. The team, then can pull up all auto coded text related to personal characteristics, for example, from across the entire data set and examine it at once. This allows a more consistent look at the related data that can then evolve more quickly, allowing the team to identify patterns that occur across data.

Auto coding is a useful approach, but has its drawbacks. One of the strengths of the coding approach is that codes can overlap - individuals may discuss several issues together and researchers can layer multiple codes together. Not only does this help to give a true characterization of the data, but common patterns in overlaps may provide useful insights. For example, the proficiency of big picture thinking was often discussed simultaneously with several of the values that systems engineers provide. This helped explain, for example, the relationship between big picture thinking as a critical skill and how that approach can provide value on diverse teams. But when using auto coding, layered codes are not possible; in the example of big picture thinking and value, the text would be tagged either as "Personal Characteristics" or "Proficiency" - not both. Since auto coding is only the first step, there are additional opportunities to create the layering and complexity that reflects the nature of the data. However, auto coding does limit the researcher to make a choice about what is most important or most prevalent in a section at the outset, which raises the risk that important relationships could be missed later. The other drawback to auto coding is that categories had to be developed and applied to all data and, therefore, could not happen early in the project. The team agreed to a limited set of categories, largely aligning with elements of *Atlas 0.25* published in 2014.

If auto coding was not used – for example, if there were a new area of inquiry, meaning that no headings had previously been identified and applied – then the team had to manually review and code all ~6,000 pages of data. Though keyword searches could be used, there was a risk that data could be missed if only keyword searches were used. For this reason, the team used a variety of keyword searches related to a given topic as well as a scanning read of a transcript when doing the initial pass for manual coding. For example, when looking for information on training, keywords included, "train," "course," "class," "learn", and "study". Once the initial coding was complete, this is essentially equivalent to auto coding in terms of level of depth.

Additional codes were then added to this subset of the data to further clarify the patterns. For example, a total of 30 individual personal characteristics were identified by participants. Some of these, in the discussion, were directly linked to the values that they helped to provide – these sections were double coded for both the characteristic and the value. Once all of the data had been analyzed, the team identified a reportable threshold – for example, for personal characteristics there were 141 excerpts and 30 characteristics.

Individual characteristics were mentioned anywhere from 15 times to only a single time across the excerpts. While none of the characteristics is "wrong", it was also not useful to simply provide a laundry list of items, particularly those that were only mentioned once across such a large dataset. It was more useful to first identify whether there were any relationships between items that might help identify areas of importance. This was done by comparing overlaps between codes. In other words, a single excerpt might be coded for multiple characteristics that were discussed together. By examining how often characteristics were cross-coded, it helped to identify relationships that participants believed are important across organizations. For example, in terms of personal characteristics, ambition and internal motivation often were discussed simultaneously, which is why they are grouped together in *Atlas*. Figure 3 provides an example of the coding comparisons conducted by the Helix teams. The higher the bars, the higher the overlap in coding between characteristics. This provides Helix with insight into relationships between and patterns around characteristics based on how interviewees discussed them.

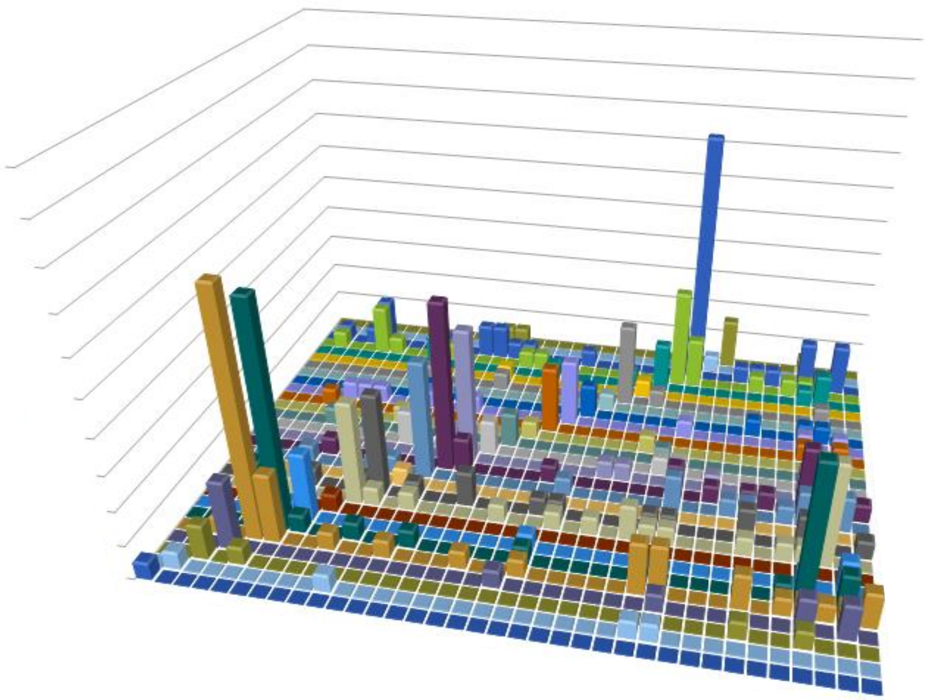


Figure 3. Example of Coding Relationships

When reporting, there was also a natural cutoff – again, in the example of Personal Characteristics, any characteristic mentioned in more than 6% of the excerpts. This was a natural threshold because the next cluster of characteristics occurred in approximately 1% or less. Items above the 6% threshold were reported in *Atlas 0.5*; items below it were not. However, because the coding structure remains in place, as additional data is added, if other characteristics become more prevalent, then they may be added to the next iteration of Atlas. Personal Characteristics were also presented in the descending order by the number of excerpts in the dataset; e.g., "self awareness" is listed first because it was discussed in the highest number of excerpts.

2.3.2 Tool Support and Combining Qualitative Analysis with Demographics

To support the its analysis, the team has imported all of the data into NVIVO (QSR International 2016), a powerful and popular commercial qualitative analysis tool. NVIVO allows the team to code text as well as overlay additional information and identifiers about the sources. This replaces tools used early in the project, namely a combination of Dedoose (Dedoose 2016) and Microsoft Excel, which proved insufficient to handle the volume and diversity of data required. In NVIVO, the sources are tagged with a code for the organization in which the individual worked at the time of the Helix interview, and then each organization is linked to characteristics such as whether it is government or commercial; whether defense, healthcare, transportation, etc.; and how systems engineer are organized, e.g. embedded, matrixed, etc. As much as possible, each comment is also tagged for the individual who made it, though in the summaries from the earliest interviews, there are a few exceptions. Each individual, again, has several characteristics, including gender, whether they are a systems engineer or a peer, and some results of the Helix career path analysis.

3 CAREER PATH METHOD AND VISUALIZATION

The career path visualization method presented here supplements the qualitative data analysis described earlier with more quantitative information about an individual's career. This analysis was conducted for the 181 systems engineers[2] from a dozen organizations who had participated in Helix as of December 2014. The initial data collection for career analysis was conducted by:

- Reviewing the resumes submitted by each individual, including chronology, organizations, position titles, and all descriptive text provided within the resumes;
- Reviewing interview transcripts and notes to add detail to the resume data;
- Reviewing the preliminary results during follow up interviews to clarify analysis. Individuals self-selected whether or not they would like to participate in follow-up interviews; roughly half of the individuals in the career analysis sample have participated in follow-up interviews; and
- Comparing the career paths with existing Helix research on the proficiencies of systems engineers and how career path elements may relate to these proficiencies. (Pyster et al. 2014b)

Using this approach, the Helix team developed a method to examine experiences and a common framework to capture, analyze, and visualize career paths. The self-assessment tool(s) provided to individuals participating in Helix to create their own career paths, including their proficiencies over time, are available in Appendix A.

3.1 CHARACTERIZING A SYSTEMS ENGINEER'S EXPERIENCES

Experimental literature on experiences has primarily focused on two metrics for experience: time (e.g. Ford et al. 1993; Schmidt et al. 1986; Firth 1979; Davidz 2006) and the frequency of times a specific task or activity of interest was performed (e.g. Stuart and Abetti 2002). Additional literature classifies human subjects based on their experiences – which is subtly different than classifying the experiences themselves – often using a combination of time and the frequency of tasks performed. This approach may also include considerations for specific roles played (e.g. Stuart and Abetti 2002, Kor 2003, Kirschenbaum 1992, Broome et al. 2001). Additional literature in the field of systems engineering, such as Sheard's "Twelve Systems Engineering Roles" (1996) or the *Graduate Reference Curriculum for Systems Engineering (GRCSE)* (Pyster et al. 2012) indicate, though, that the characterization of experiences is critically important to understanding how experiences enable growth.

The first challenge was to determine a common "unit of measure" for experience. Though time is common, it was not easily used in the data available. For example, if someone described a position they held over a five-year period, they did not explain the portion of time taken up by the activities they performed over those five years. In addition, several individuals submitted information on their careers that included detailed descriptions, but did not include markers for chronological time. Because of these data limitations, the Helix team chose to use a **position** as the unit of measure for experience.

---

[2]The interviews for all systems engineers in the Helix study are included here. However, the resume data was not provided for some of these individuals, and not all resume data was sufficient to complete each type of analysis. In general, the career analysis sample is N=157. Where the analysis looks at a subset of the sample or where individuals were eliminated from analysis for insufficient data, the sample size (N=x) is provided in the text.

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

- **Relevance:** A 'relevant' position is one that enables a systems engineer to develop the proficiencies critical to systems engineering.
- **Position:** Every systems engineer who is employed at an organization fills a position that is established by the organization; that organization also defines the roles and responsibilities to be performed. Helix considers position as a 'unit of measure' for experience, since most of the characteristics of experience are in the context of the position that is held. A 'systems engineering' position is one where the individual's primary focus was on systems engineering activities.
- **Chronological Time:** The amount of time spent in any particular position or in performing a role.
- **Number of Organizations:** The number of different organizations that an individual has worked at, not counting internal movement within an organization across departments or divisions, reflects the variety of types of experiences that one may possess.
- **Organizational Sectors:** There are many differences in the general characteristics of an organization based on its sector. In *Atlas*, three organizational sectors are identified: government, industry, and academia.
- **Roles:** A role is a collection of related systems engineering activities. Roles were identified based on the activities consistently performed by systems engineers. There are 16 roles identified in *Atlas*, as described in Section 3.5, below.
- **Lifecycle Phases:** Generic systems engineering lifecycle phases considered in *Atlas* are based on the lifecycle phases in the *Guide to the Systems Engineering Body of Knowledge (SEBoK*). (BKCASE Authors 2015)
- **Systems:** There are many aspects to the types of systems on which a systems engineer could work. Working across these different categories provides valuable experience to an individual systems engineer.
    - **Domain:** This is the primary area of application for the systems being worked on. However, there are many domain categorizations; some domains also relate to industry sectors.
    - **Type:** Product systems, service systems, and enterprise systems are three major types of systems, depending on the nature and composition of the system of interest. System of systems is another paradigm in systems engineering, and could be a combination of one or more types of systems.
    - **Level:** A systems engineer could work on various levels of a system: component/element, subsystem, system, and platform or system of systems.

By using the data available for each individual, the characteristics of each position played and the order that they played them can be identified. Looking for patterns across the Helix data set, this information can be used to develop a preliminary understanding of how career paths shape proficiency.

The ways in which positions were categorized were pulled from existing literature wherever possible. For example, a systems engineer working in the commercial sector of a company may define life cycle in different terms than those used by a US Department of Defense systems engineer. To normalize the discussion, the definition of life cycle stages from the *Guide to the Systems Engineering Body of Knowledge* (SEBoK) was used; the interviewee's own words and phrasing were compared with the descriptions of life cycle stages in the SEBoK and categorized appropriately. (BKCASE Editorial Board, 2014) Likewise, the roles played by the interviewees were based on Sarah Sheard's "Twelve Roles of Systems Engineers" (Sheard 1996), although roles have been added to reflect what was seen in the data. Where existing literature was not available, categories were created that reflect the character of the data.

3.2 Characterizing a Systems Engineer's Education

Education plays two key roles in the development of systems engineers. First, it provides the foundation knowledge to support engineering-related work. Typically, this takes the form of undergraduate education in an engineering discipline, technical field, or physical science. Second, graduate level education is an avenue to develop more advanced skills, explore more in-depth knowledge, and help systems engineers grow as they move through their careers.

The characterization of education was much more straightforward than the characterization of experiences. For each systems engineer in the sample, the team recorded:

- **Chronological Time:** The date of the completion of the degree program.
- **Type of Degree:** This is the level of education an individual achieved. The categories used were: bachelor's, master's, and doctor of philosophy (PhD). For this analysis, only education that resulted in a degree was recorded. Individuals did receive graduate certificates or took individual courses, but there was not enough data to draw any meaningful conclusions. Also, if a degree was in progress but not completed, it was not recorded.
- **Field of Study:** The primary discipline on which the individual's education was focused. These were initially recorded as reported. Over time, categories of related fields of study were created.

All systems engineers in the Helix sample held at least a bachelor's degree and the majority – 58% – held at least a master's degree.

3.3 Identifying Key Positions

A third aspect of career paths are the key milestones for a systems engineer's career. The Helix team focused on major steps or changes in a systems engineer's positions. A ***position*** is equivalent to the roles and responsibilities associated with an individual's title. Organizations will define what roles and responsibilities each position contains and *position* descriptions may not translate across organizations. The key positions identified for systems engineer included:

- **First systems engineering position.** This was self-identified by participants as the first position in which systems engineering responsibilities were the *primary focus* of a position, though they may have non-systems engineering responsibilities as well. This was often difficult to identify, because participants indicated that their roles often transitioned gradually and it was hard to identify when they officially became systems engineers, especially because so many never had that specific title. The Helix team recorded this information in whatever way it was provided by participants. In a few organizations, the hierarchy and structure for becoming a systems engineer was much more well-defined, and for individuals in those organizations, the transition to systems engineer was more easily identified.
- **Chief systems engineering positions.** A chief systems engineer (CSE) is someone who has formal responsibility to oversee and shepherd the technical correctness of a system, often coordinating with many other systems engineers who have smaller scopes of responsibility. These milestones

are any positions in which an individual acted as a CSE, *regardless of their title within their organization*.

- **Project manager positions.** A project manager is someone who has formal responsibility to oversee the programmatic aspects of a system, generally focused on budget and schedule. Project management responsibilities sometimes overlap with SE responsibilities, particularly those around planning and management; in some instances, a CSE may also function as a PM.

These milestones are important for understanding how the nature of a systems engineer's work has changed over time. It also gives insights into how quickly an individual progresses through different stages of her career. By comparing these patterns across individuals, common ranges of progression can be identified, as can outliers. For example, among the CSEs discussed in this paper, one individual became a CSE only 8 years after completing his undergraduate degree. However, only 12% of CSEs gained their first CSE position within 10 years after entering the workforce; therefore, this is an outlier rather than typical for for CSEs.

3.4 Assessing Proficiency

Interviewees were asked about not only their common activities but what they believe were the critical knowledge, skills, abilities, behaviors, and patterns of thought (cognitions) that enable them to be effective in performing those activities. Helix calls these **proficiencies**.

By coding all of these responses individually and then aggregating like responses, the Helix team has identified the key proficiencies of systems engineers. These are elaborated in (Pyster et al. 2015). In brief, there are six proficiency areas, each of which contains several related groups of skills, or **categories**, as described below:

1. **Math/Science/General Engineering:** Foundational concepts from mathematics, physical sciences, and general engineering. Categories include: Natural Science Foundations; Engineering Fundamentals; Probability and Statistics; Calculus & Analytical Geometry; and Computing Fundamentals.
2. **System's Domain & Operational Context:** Relevant domains, disciplines, and technologies for a given system and its operation. Categories include: relevant domains, relevant technologies and systems; relevant disciplines; familiarity with the system's concept of operations.
3. **Systems Engineering Discipline:** Foundation of systems science and systems engineering knowledge. Categories include: lifecycle; systems engineering management; systems engineering methods, processes, and tools; and system complexity.
4. **Systems Engineering Mindset:** Skills, behaviors, and cognition associated with being a systems engineer. Categories include: big-picture thinking; paradoxical mindset; flexible comfort zone; abstraction; and foresight and vision.
5. **Interpersonal Skills:** Skills and behaviors associated with the ability to work effectively in a team environment and to coordinate across the problem domain and solution domain. Categories include: communication; listening and comprehension; working in a team; influence, persuasion, and negotiation; and building a social network.
6. **Technical Leadership:** Skills and behaviors associated with the ability to guide a diverse team of experts toward a specific technical goal. Categories include: building and orchestrating a diverse team; balanced decision making and rational risk taking; managing

stakeholders and their needs; conflict resolution and barrier breaking; and business and project management skills.

In addition, the non-systems engineers in the sample – project managers, classic engineers, executive leadership, and human resources personnel (HR) – were asked which proficiencies they considered critical in the most effective systems engineers with whom they worked. These were also coded and aggregated with the systems engineers' responses; they validated the existing categories.

In 2015, the Helix team provided the and reviewed the draft proficiency model to participants and had them react to the categories and structure directly. The existing structure was validated, with no additional skills being cited that did not fit within existing categories; this did, however, help the team in re-allocating some proficiencies to other categories to make them more easily understood by a wider audience.

Finally, systems engineers were asked to perform self-assessments of their own proficiencies at different points in their careers, which could then be overlaid with their career paths. Early on, the Helix team would perform its own assessments during these discussions and map them against the self-assessments to ensure alignment between the team's approach and the participants'. They were also asked to cite what they believed were the most critical proficiencies for their *current* positions. In addition, some were asked to identify what they believed were the minimum proficiencies to be effective in their current positions. Non-systems engineers also did self-assessments, to help identify where these proficiencies overlap with other disciplines. In addition, they were asked what they believed were the most critical proficiencies or the minimum proficiency level they would desire in the systems engineers that they work with. All of this work helped to validate the proficiency set as a useful and comprehensive model.

The forces identified in Figure 1 – experiences, mentoring, education and training – are linked to the growth of proficiency by interview data. When an individual would cite a critical skill, the Helix team would ask how that individual had developed that skill over time. These types of discussions were cross-coded for both the relevant force(s) and the related proficiency(ies). Again, the self-assessment tool for proficiency can be found in Appendix A.

3.5 Mapping a Career Timeline

As described above, chronological time is an attribute of all positions. The final step in developing a career path map for an individual was to create a visualization over time of all of the elements listed above. This visualization lays out all of an individual's positions, and their characteristics over time, with their education, the career milestones, and their proficiency assessments. Figure 4, below, shows a generic example of this.

In Figure 4, only the timing, roles and the lifecycle stages characteristics of positions are illustrated. This is for two reasons: one is ease of visualization in a single graphic – though any combination of attributes is possible in this format – and the second is that systems engineers were able to provide the clearest discussions on how *roles* and *lifecycle* exposure contribute to proficiency. For other attributes, these relationships were more sporadically represented in the data; in addition, not all systems engineers provided basic data on all attributes, but the Helix team was able to complete roles and lifecycle data for nearly the entire dataset (93% for roles; 91% for lifecycles).

By creating these individuals "maps" for each career path, it is possible to start identifying patterns – not only in proficiencies but in the common attributes that lead to similar proficiency profiles. Additional analysis of the career paths of individuals in similar roles was also insightful; even though there is some individuality to each systems engineer's career, the common patterns indicate ways that systems engineers may typically grow – or areas where certain types of systems engineers differ from others. The analysis highlighted in this paper is that of chief systems engineers – not only of their career paths overall, but in a few critical cases, highlighting their differences from other senior systems engineers.

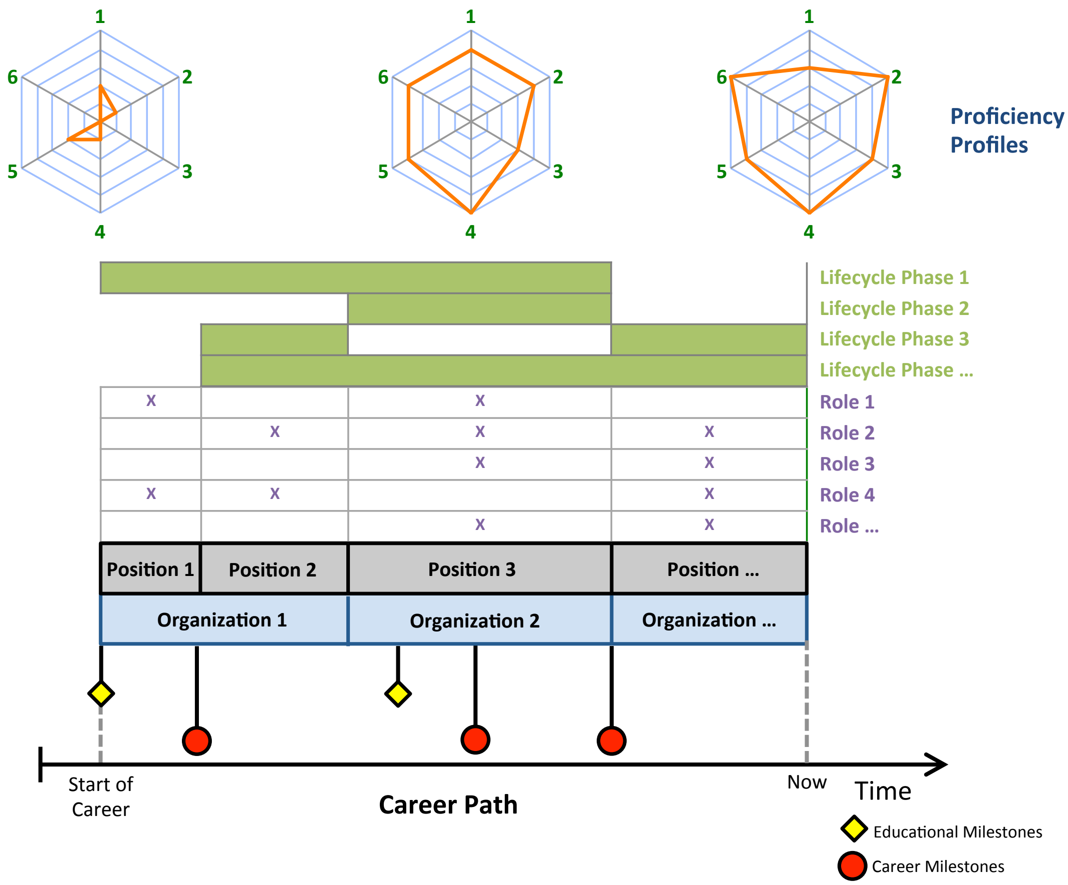


Figure 4. Generic Career Path Mapping

3.6 Steps to Validate Findings from Career Paths

There were several ways in which the findings from career paths were verified and validated. The most straightforward was verification through follow-up interviews, where individuals were presented with the current analysis of their career paths and asked to provide any corrections or fill any gaps. Though only 48% of interviewees also participated in follow-up interviews, the changes and updates were minimal, and often reflected additions where certain aspects of an individual's career had not been discussed in the initial interviews. Because the reviews by individuals of their own career paths revealed no major issues with the methods, the Helix team considered that the method is a valid approach to understanding the causes of change and growth over time.

In terms of additional validation, the Helix team acknowledges that, for a number of reasons, the way an individual progressed through her career may not have been an optimal approach. Participants were asked to identify areas where what they would have approached their careers differently based on their current levels of insight. They were also asked their general satisfaction with their career progression and to identify areas where colleagues might have had a different, preferred approach. And because – as explained in the dataset discussion above – the interviewees identified themselves and their organizations identified them as all of "average" to "excellent" effectiveness, it is reasonable to draw conclusions about the career paths of these individuals.

Finally, in 2015, the Helix team worked with one organization to pilot the career path approach. Individuals worked through an example career path with the Helix team and then mapped their own careers in real time. The feedback was that this approach was much better structured and focused than any career guidance they had received. In some cases, this reinforced that their current planning was appropriate, and other individuals reported that with insights from their career paths, they realized that they needed to seek new opportunities. All of the participants (n=34) agreed that this approach was useful and would be a valuable tool for them as individual systems engineers and for the organization.

4 ADDITIONAL ANALYSIS ON ELEMENTS OF ATLAS

Atlas 0.6 is an ***incremental*** improvement based on additional data analysis of the Helix data set reported in Pyster et al. (2015a and 2015b). The team conducted additional analyses to provide insights into areas of *Atlas* where insights in version 0.5 were less mature.

The overview of *Atlas* in the context of an individual systems engineer employed in an organization is captured in the systemigram illustrated in Figure 5. A systemigram consists of nodes that contain noun phrases, links that contain verb phrases, and is to be read as sentences along the direction of the arrows. The primary sentence is read from the top left node to the bottom right node and presents the main theme of the systemigram. In the ensuing discussions, sentences to be read in the systemigram are italicized, where nodes are represented in square brackets.

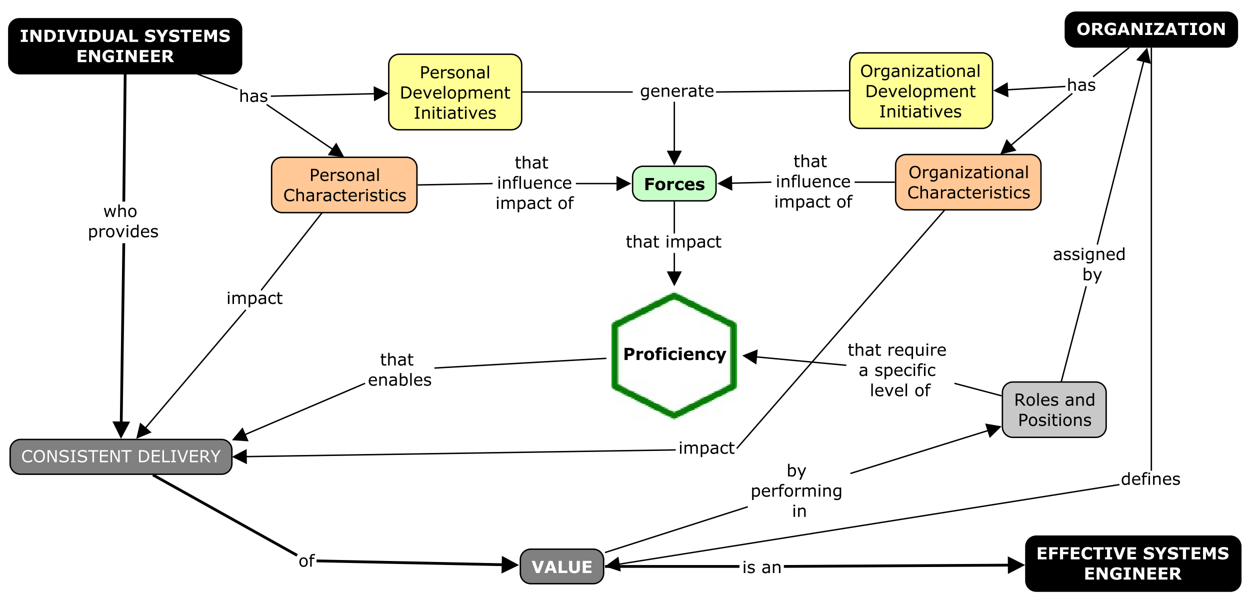


Figure 5. *Atlas* Overview

From Figure 5 above, it can be seen that the main theme of *Atlas* is: '*[Individual Systems Engineer] who provides [Consistent Delivery] of [Value] is an [Effective Systems Engineer]*'. This fundamental definition of an effective systems engineer hinges on *[Value]*, and it can be seen that '*[Organization] defines [Value]*'. Therefore, it is on the organization to define the value that the systems engineer is expected to provide. Further, the individual systems engineer provides '*[Value] by performing in [Roles and Positions] assigned by [Organization]*'. Therefore, it is again on the organization to establish the position of the systems engineer in terms of roles and responsibilities, keeping in mind that '*[Positions] require a specific level of [Proficiency] that enables [Consistent Delivery] of [Value]*'.

The core of *Atlas* is the proficiency of the individual systems engineer - what it is, and how it can be improved. '*[Individual Systems Engineer] has [Personal Development Initiatives]*' and '*[Organization] has [Organizational Development Initiatives]*'; together, they '*generate [Forces] that impact [Proficiency]*'. At the same time, '*[Individual Systems Engineer] has [Personal Characteristics] that influence the impact of [Forces]' and '[Organization] has [Organizational Characteristics] that influence the impact of [Forces]'* –

these forces may have a positive or a negative influence. Further, both personal enabling characteristics and organizational characteristics '*impact [Consistent Delivery] of [Value]*'; again, the impact can be positive or negative. Amidst all these influences and impacts, the challenge for the individual systems engineer and the organization is to improve the '*[Proficiency] that enables [Consistent Delivery] of [Value]*' to the organization.

For *Atlas 0.6*, the team focused analysis efforts on the following areas:

- *Training.* The forces of Experiences, Mentoring, and Education were described in detail in *Atlas 0.5*.
- *Personal Development Initiatives*. These are the ways that individuals seek to improve their own proficiency *outside of organizational initiatives*. In other words, ways that individuals seek to improve their proficiency other than by participating in organization-sponsored activities and initiatives
- *Organizational Development Initiatives.* These are sponsored approaches designed to improve the workforce in some way.

4.1 TRAINING

Training, a primary way that organizations attempt to improve the skills of their workforce *en masse,* was consistently discussed by interviewees. Interviewees did not provide a specific definition of training, but the following definition encompasses the ways training was described: organized activity aimed at imparting information and/or instructions to improve the recipient's performance or to help him or her attain a required level of knowledge or skill (Business Dictionary 2016). The discussion of training below reflects uses of the term training, which fit this perspective.

There were three additional ways in which the term "training" was used: as an undifferentiated way of describing education; as a way of describing the overarching capabilities of an individual; or in the term "on the job training". For example, several individuals discussed "training" when referring to degree programs at a university. According to the Helix definitions, this falls under "education", and therefore these instances were coded as part of education instead of training. Examples of the second alternative use of the term include phrases such as, "a mechanical engineer by training" or "I do systems engineering work but I am not trained as a systems engineer." Generally, these same individuals would then describe the organizationally-sponsored training they had received in systems engineering; therefore, in these instances the term "training" or "trained" referred not to specific instances of training but really the overarching career path of the individual. These instances were coded as referring to the career paths of the individuals.

"On the job" training was commonly cited, but nearly 100% of the time this was a euphemism for learning through experience and *not* an actual training program or approach. In these instances, these excerpts were coded for experience. Occasionally this referred to an apprenticeship-type arrangement where the individual was partnered with a more senior person who was more intentionally guiding learning and development. In these instances, this was a blend of training, experiences, and mentoring and was coded accordingly.

Three main aspects of training emerged from the data: types of training; subject matter; and common issues or best practices. Each is described in a subsection below. The analysis is based on 647 excerpts around training from across the complete dataset.

4.1.1 Types of Training

The most common characteristics for the different types of training were:

- Delivery mode – this describes the mechanisms used to provide information to the individual(s) who participate in training. Most common delivery modes discussed were in-person and online modules. There were some self-study modules that were paper based.
- Length – the chronological time allocated to training activities for a particular training activity. The most common times were either less than a day; 1 day; 2 days; 5 days; or 2 weeks.
- Organizational push versus personal pull. This is whether an organization mandates or strongly encourages the training or whether the training is specifically sought by an individual.

These things in combination determine the overall character of the training and there were some combinations of training that were more common than others. The subject matter of the course (see section 4.1.2) sometimes dictates the delivery mode and length, but just as often organizational constraints will impact these factors. Likewise, the Organizational Characteristics will influence whether the training is pushed out to individuals or whether individuals must actively seek opportunities to attend training.

Some common types of training discussed in multiple organizations include:

- **In-person short courses.** These were considered "short" courses because they were chronologically short efforts, unlike semester-long courses. These types of courses occurred in 100% of organizations. Generally, these were 1-, 2-, 5-, or 10-day courses, with 1- and 2-day formats being the most common.
    - 1- or 2- day short courses tended to focus on more discrete subject matter. For example, a particular approach to architecture, an introduction to a specific tool, or teaching points on a particular skill like communication.
    - 5-day or longer courses tend to be for broader training. For example, several organizations had an "introduction to systems engineering" course that lasted a week or more. In-depth leadership training also tended to be involved in the longer format courses.
- **Online training.** Generally this type of training focused on very discrete skillsets and was limited to a few hours. For example, a tool tutorial or an overview of organizational process would be common subject matter for this format. These types of training were usually developed internally. Most organizations had some online component to their training, though all participants who discussed the issue consistently stated that they found organization-sponsored in-person training more effective than organization-sponsored online training.
- **Seminars.** "Lunch-and-learns" or "brown bag" training courses were another common approach. These are informal short sessions, usually no more than two hours on very specific topics. Typically, a subject matter expert (SME) on the area within the organization will generate and provide the training. Participants indicated that these are informative and useful, but an individual session does not create a major change in proficiency. This type of training was reported in 40% of the organizations.

The language to create the categories above comes from the data, but aligns with delivery modes described in the literature (e.g. Buch and Bartley 2002) with the exception of the seminars. These types of seminars were more commonly grouped with in-person (or classroom based) instruction, but because the subjects of the Helix data view the purpose and outcomes of these types of training differently, they are separated here.

There was a broad mix of "mandated" versus "available" training, though in 90% of organizations there were occasional to consistent refrains that training was not as available as it should be. Even mandated training was sometimes difficult to obtain due to organizational or project constraints. For example, if an individual was deemed "too valuable" on a project, she may be told that the project could not allow her time off to attend even mandatory training, or a mid-level systems engineer might be told by his manager that "mandatory" training was more critical for more junior systems engineers and, therefore, he would not be able to attend. Participants consistently stated that the training they needed was not fully available and that they had consistent support to attend any training they needed in only 10% of the organizations.

4.1.2 Training Subjects

Training can cover a wide variety of subjects and not all of them are related to systems engineering. For this analysis, the Helix team coded only training that was related in some way to the *Atlas* proficiencies. For clarity, the common training subjects reported below are grouped into the proficiency area to which they most closely relate.

- **Systems Domain and Operational Context.** There were some training subjects that addressed elements of the domain(s) relevant to the system and the context in which the system will be used. In general, there was less coverage/conversation in this area than in others.
  - **Engineering Disciplines.** A few organizations provided introductory courses for systems engineers regarding other engineering disciplines, helping to identify "what systems engineers need to know" about software engineering, for example. These were generally one- or two-day courses focused primarily on orienting systems engineers to the terminology, key issues or concerns, and perhaps processes that are more common in the discipline. For example, in a software orientation course, agile methods – which are in many ways antithetical to the way large-scale defense systems are created – might be discussed or a comparison of object-oriented software architecture and functional architecture might be made. Again, this is generally enough to help systems engineers begin to "speak the language" in a less familiar discipline. In Bloom's Taxonomy terms, this would be to help individuals get to the "Knowledge" level, with perhaps "Comprehension" in some basic areas.
  - **Relevant Technologies.** A few organizations provided introductory courses for systems engineers regarding specific technologies that are critical to the systems at hand. These were generally one- or two-day courses and similar to those on Engineering Disciplines, focused primarily on orienting systems engineers to the terminology and key issues or concerns for the technology of interest. For example, in an organization that focused primarily on communications, there were short courses on radar technologies, which included the basic terminology, the basic physics of how radar technology works, and critical considerations for radar technologies in development. Again, in Bloom's Taxonomy terms, these courses would be to help individuals get to the "Knowledge" level, with perhaps "Comprehension" in some basic areas and are generally targeted to people who are experienced in other domains, but new to the system of interest.
- **Systems Engineering Discipline.** Perhaps unsurprisingly, one of the most common types of training discussed was training on systems engineering itself, which was offered in some form by

each of the organizations participating in Helix, though junior and mid-level systems engineers were more likely to have attended formal training than senior systems engineers.

- **Foundations.** There were several organizations (~40%) that provided a foundational or introductory "course" in systems engineering. This was often a multi-day course (sometimes up to a 5-day course) that covered the basics of the discipline, provided an overview of the systems lifecycle, introduced terminology, the value of systems engineering, etc. These courses tended to be targeted to new systems engineers transitioning from other disciplines (such as classic engineering) or younger systems engineers who had relatively little experience. Typically, mid-level and senior systems engineers stated that this type of course would not be useful for them because a one-week training course could not match what they had learned from their wealth of experience; however, they often felt it was useful to individuals new to the discipline.
  - Most often, these courses are developed internally and also include "branding" of the processes and methods for systems engineering within the organization, as opposed to being based on a wider community view such as the INCOSE *Systems Engineering Handbook* (INCOSE 2015).
  - In some organizations (15%), there is a shorter version of this course (for example a half-day or one-day course) that is provided to executive leadership, management, project managers, etc. In organizations where this approach was used, interviewees tended to report a more supportive culture or at least reported that the value and acceptance of SE in the organization was improving.
- **Specific Topics.** In some organizations, only the overview SE course was offered. However, other organizations offered a variety of training courses in SE-related topics. Some common examples included architecture, risk management, decision management, etc. These tended to be shorter courses than the overview courses (often 1 or 2 days) and might include general terminology and knowledge, but would also include the specific approaches used by the organization and perhaps an introduction to any critical tools used by the organization.
- **Tool-Specific Training.** There were a number of training courses offered around specific tools that support systems engineering work. Again, these tended to be short courses that were very focused on the *use* of the tools and not on the rationale behind the tools. For example, DOORS (IBM 2016) was the most commonly-cited tool and the most-commonly mentioned tool-specific training. But training was described as "how to use DOORS" and not on how to elicit good requirements. Other common tools were architecture-focused tools, MindMapper (MindMapper 2016), and tools to support model-based systems engineering. Overall, this was generally viewed as useful training and in some organizations where the tools were available but *not* training courses, this was listed as a critical gap. However, individuals stated that if they did not immediately use the tools on the job, the training became less effective, as those skills would be quickly lost.
- **Systems Engineering Processes**. Sometimes incorporated into a systems engineering overview course, this was also occasionally offered as a stand-alone course. This type of training would specifically help individuals understand the processes used to conduct systems engineering *in their organization* and may or may not have ties to systems engineering processes such as DoD 5000.02 (DoD 2015). A common shortcoming cited

with this type of training was that it trains the process steps but does not explain why the process looks the way they do and does not teach how to tailor the process; both shortcomings were cited in several organizations as areas in which this training could be improved.

- **Interpersonal Skills.** Coverage of training for these types of skills was mixed. Where it was available, this type of training was generally seen as high value in helping develop critical skills for systems engineers that might be underdeveloped in other engineers.
  - **Communication.** Several organizations provide a variety of training designed to improve communications skills. This could cover interpersonal communication, writing, or presentation. Overall, systems engineers tended to report training on interpersonal communication – particularly on how to communicate effectively with diverse groups of people – to be very valuable. Training on writing was cited as most useful when it was on technical writing that specifically supported the systems engineers' work. Presentation training was rare, but valued when available.
  - **Teamwork.** A few organizations reported that training was available that really focused on interpersonal relationships and how to work in a team. Among other topics were creative abrasion and bringing a diverse group to solution. Again, where offered, these types of training were seen as valuable.
- **Technical Leadership.** In the area of technical leadership, as with Interpersonal Skills, training was not available in all organizations but was generally valued when it was available.
  - **Leadership training.** Individuals in 50% of the organizations in the sample reported some form of leadership training. In 15%, this was associated with "high potential" programs – often rotational programs that are designed to grow systems engineers – and so are offered to a limited number of people. But in most, there are multiple leadership training opportunities available, though again the approaches vary between organizations. Some have basic modules that can be studied online to deal with specific topics like conflict resolution while others have broad courses available. In a few organizations (10%), there is tiered leadership training that goes from a one-day course for all systems engineers to a one-week course for more senior individuals to a two-week course for the most senior leadership. This training was typically reported as valuable by the individuals who had participated.
  - **Team building.** As opposed to working in a team, only 5% of organizations reported providing training that actually focused on how to bring a team together and lead a team, versus simply being a productive member of a team.
  - **Related Disciplines.** Technical leadership overlaps with relevant skills from other disciplines, particularly project management and business. In some organizations, systems engineers are encouraged or allowed to take training courses in these areas. Project management courses are more readily available, but business courses, which are typically tailored to the business aspects relevant to the context of the organization, were seen as extremely useful.

Though the utility of each type of training varied between organizations, and indeed occasionally between individuals within the same organization, overall individuals were pleased with the opportunity to attend training and learn new things.

In general, the data was not detailed enough to draw specific conclusions about the change in proficiency expected for training in specific subjects. However, among the 97 individuals who performed proficiency self-assessments, training was described as accounting for at most an improvement of 1 or 2 on an ordinal 1 to 10 scale. In other words, if an individual rated herself as having grown in proficiency in an area from "2" to "7", she would not generally attribute more than 1 or 2 ordinal steps in proficiency growth to training. Training can provide some improvements in proficiency, but were in no way considered a substitute for experiences. And importantly, of the individuals who provided insight on best practices, 67% indicated *actually applying learning from training during or soon after a training event is a best practice* and 17% of the issues with training raised indicated that if an individual did not apply the learning from a training program on the job immediately or soon after the training, then the impact of the training could be lost. This is consistent with training literature which shows that training transfer – the ability to apply skills learned in training on the job – is limited when individuals do not have opportunities to use new learning in their work (e.g. Brinkerhoff and Montesino 1995, Lim and Mooris 2006, and Burke and Hutchins 2007). Clarke (2002) states that limited opportunity to perform skills on the job is the primary limiting factor on training transfer.

4.1.3 Training Issues and Best Practices

There were 103 excerpts in which individuals discussed issues, best practices, or risks. Among these excerpts, there were a few issues identified consistently by participants.

- **Training must be "immediately" applied or lost.** Participants indicated training that could be applied on the job during or immediately after training was internalized and remembered much more effectively than training which occurred "in the classroom" only, which was much more likely to need to be repeated. Likewise, in the discussion of proficiency changes over time, individuals indicated that training that was applied resulted in a *long-term* increase in proficiency while simply attending a training course would provide a *temporary* increase in proficiency that would diminish without application. This aligns with findings from the literature about the importance of the "opportunity to perform" (Burke and Hutchins 2007) for training transfer.
- **Access to training.** Across government and commercial organizations and across a variety of domains, a common issue that was cited was access to training. The most common issue was that even when training was mandatory or when it was optional but cited as a priority within the organization, it was often difficult to gain access to training. There were a variety of reasons, but as participants discussed the issues, they came down to a few common causes: a lack of funding for training; a disconnect between organizational and project priority; and lack of managerial support. Particularly in government organizations or commercial organizations in the defense industry still recovering from the 2012-2013 sequestrations, the lack of funds for training was a very common concern. There were many individuals who stated that while the leadership of the organization encouraged training, project managers balked at allowing individuals leave time to attend even partial-day training. Likewise, several individuals stated that while they would request training, their managers would not support their attendance, though in some instances this was tied to lack of available training funds. A related issue was the dissemination of information about training opportunities within an organization. In other words, individual systems engineers were not always aware of the training courses available, even though the Helix team was informed of these initiatives by the leadership team.

- **Some anticipated risks can be addressed by training.** These included how organizations could become more agile, better incorporate and integrate COTS into design, and MBSE were topics that were mentioned. Common themes were the lack of domain knowledge and the rationale behind the processes used within organization. The former would be a way to help younger systems engineers who would have to mature more rapidly to replace retiring senior systems engineers. For the latter, there were many training opportunities on process, but a common criticism of them was that while the process *steps* are taught, the rationale behind the process – how it came to be, why and when it might be tailored – was not understood consistently, particularly among junior systems engineers.

4.2 Personal Development Initiatives

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

- *Individual Reading* – Some individuals reported that they spent personal time on reading that was related to their work. This included a variety of things, such as journal articles, conference papers, trade publications, relevant news or magazine articles, or books. Journal articles, conference papers, trade publications, and new articles tended to be around technical subjects – new technologies related to the systems the individual supported, classic engineering disciplines, relevant domains, or systems engineering itself (such as the *Systems Engineering* journal or the IEEE *Systems* journal). In terms of books, however, these were more commonly around non-technical areas such as technical leadership, particularly business, or interpersonal skills, particularly communication.

- *Attending conferences* – several individuals stated that they attended conferences relevant to their work whenever possible. This could be a mix of domain-specific, classic engineering, systems engineering, or project management related. Individuals who attended conferences stated that their organizations sponsored their attendance but that this was not a broad initiative; rather, their individual managers or programs helped them find funding to attend relevant events. An additional few individuals indicated that they used to attend conferences, but that funding was no longer available for these efforts and had not been for the last five years or more.

- *Online courses* – these are not full academic courses for credit that could be counted towards a degree. Those types of courses were considered education. However, a few individuals indicated that there were free courses available online, for example massive open online courses (MOOCs) or small, university-sponsored free courses on relevant topics. There are a variety of topics, including overviews of basic classic engineering disciplines such as electrical or software engineering, topics such as risk- or decision-management, or specific technology areas. Individuals who took these courses indicated that they were helpful for gaining an overview, particularly in areas that were relevant to the systems and individual worked on but in which he did not have experience. These courses are free, so participation in them is wholly dependent on the motivation of the individual.

- *Certification – All* of the DoD organizations required an engineering certification (at the time of the Helix interviews, the Systems Planning, Research, Development, and Engineering (SPRDE) certification) for all of their systems engineers. However, a few individuals have sought individual certification. None of the organizations specifically sponsored external certification initiatives and the few individuals who had become certified said that they did not believe that it would help them in their organizations but they felt it was important for them as individuals. The three types of certifications discussed were INCOSE Certified Systems Engineering Profession (CSEP); PMI Project Management Profession (PMP); and state-certified professional engineer, which was not specific to systems engineering.

Of the individuals who stated they did not do anything outside of organizational initiatives, many junior and mid-level systems engineers said that they would like to, but that there are roadblocks. The most common are that their work responsibilities are time-consuming enough that they do not have time for extra work or that their managers are not supportive of individuals pursuing additional training. In at least one organization, individuals stated that they were expected to pursue training but were not given leave from their roles and were "dinged on their performance" for this. Most of the senior systems engineers who discussed personal initiatives stated that beyond reading or attending conferences, they did not believe they needed to do more than continue to build upon their experiences. However, almost 5% of senior systems engineers had at one point taken the initiative to create training programs to pass on their knowledge and experiences to younger systems engineers in their organizations.

4.3 Organizational Development Initiatives

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

The discussion presented in this section is aggregated from 40% of all the Helix interviews across the 20 organizations, where organizational initiatives were discussed by the participants. In the organizations with a greater number of Helix participants, a richer view of the organization emerged, that included similarities and striking differences in the opinions of the participants. While these are highlighted in the discussion, the intent is not to provide an organization level analysis of initiatives.

4.3.1 Nature of Organizational Initiatives

While initiatives are easier to identify in a personal context, identifying initiatives in an organizational context is not always straightforward. Many features of organizational characteristics can be observed from Helix interviews.

- **Distinction between initiatives and policies**: It is not always straightforward to recognize and identify organizational initiatives, and to distinguish them from organizational practices and policies. Helix considers it an initiative if the organization plays an active role in promoting, enabling, and supporting it for the benefit of its employees. For example:
    - Some organizations provide tuition reimbursement to their employees seeking graduate degrees in related disciplines, subject to policies regarding eligibility, absence from work, etc. It is upon the individual employee and the immediate leadership (e.g., manager) to utilize those policies.
    - Other organizations play a more active role in providing graduate education for their

employees: they establish relations with specific universities; they establish cohorts for individual courses and/or degree programs; they provide facilities within their premises for the universities to conduct courses; they make available organizational data for projects and dissertations; and also tend to reward employees who go through these programs with a promotion or salary raise.

- **Scope of organizational initiatives**: While some organizational initiatives are targeted at systems engineers' proficiencies or systems engineering proficiencies or within the systems engineering department/division, there are other initiatives intended for the benefit of all employees across the entire organization, including systems engineers. On the other hand, there are also initiatives that are offered only to those systems engineers that meet certain eligibility criteria and not to the entire systems engineering population.
- **Influence of organizational initiatives on organizational characteristics:** While some organizational initiatives generate the forces that in turn improve the proficiency levels of individual systems engineers, some other organizational initiatives improve organizational characteristics – either directly or indirectly. For example:
  - Some organizations have initiatives to identify and recruit SE talent from within the organization, and also to recognize and reward achievements of systems engineers and other employees. Such initiatives do not directly improve any of the forces, but rather the organizational characteristics.
  - Some organizations have mentoring initiatives to develop their junior systems engineers by pairing them up with senior systems engineers. Such initiatives are intended to directly benefit the mentee. However, such relationships between junior and senior systems engineers also tend to improve the environment and culture of the organization.
- **Formal and informal initiatives**: By definition, organizational initiatives are formally established and deployed. However, there also exist informal versions of those formal organizational initiatives, that could even co-exist with formal versions within the same organization. Some informal initiatives are also established by the organization. For example:
  - It is typical for mentors and mentees to form an informal mentoring relationship, without being explicitly directed by the organization. Such informal mentoring relationships tend to exist irrespective of the establishment of a formal mentoring initiative in that organization.
  - Some organizations offer a variety of training courses on topics of relevance, often in a classroom setting. In addition, there are also informal training and information sessions that the organization offers – as guest lectures or lunch-and-learn programs.
- **Portfolio of initiatives**: Organizational initiatives rarely exist in isolation; typically, a portfolio of initiatives is available to employees. Organizations establish individual initiatives to address various needs; and in some cases, a higher level initiative leads to many lower level initiatives as well.

4.3.2 Types of Organizational Initiatives

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

- **Recruitment initiatives**: These initiatives are intended to recognize systems engineering talent, and to bring them into the systems engineering fold. In some organizations, such initiatives bring in new employees from outside the organization – usually fresh graduates or others with limited experience. Some other organizations have initiatives to recognize and recruit systems engineers from elsewhere in the organization.

- **Orientation initiatives**: Some initiatives are exclusively targeted at new employees to familiarize them with the organization, its processes, and the way it does systems engineering. In most organizations, a job rotation program is usually offered only to new / junior employees, offering them a glimpse into various parts of the organization before assigning them to one part of the organization. Some organizations recognize the value of such initiatives to senior employees, and extend those initiatives to them as well.

- **Experience enhancing initiatives:** Junior systems engineers grow into senior experienced systems engineers not just by the number of years they spend in an organization, but through performing in various systems engineering roles; different projects; various levels and types of systems; and different phases of a systems lifecycle. Organizations establish initiatives that are designed to effectively provide rich experiences to systems engineers.

- **Mentoring initiatives**: These initiatives are very prevalent in many organizations – either as a formal or an informal arrangement. While the primary beneficiaries of mentoring arrangements are the less experienced mentees, the more experienced mentors and the organization at large stands to benefit as well. From a Helix perspective, 'mentoring' is also identified as a force that directly impacts and enhances the proficiency of systems engineers. Various aspects of mentoring were discussed in *Atlas 0.5*.

- **Education and training initiatives:** Every employee enters any organization with some level of formal education. Recognizing the value of formal education, many organizations offer many initiatives for their employees to obtain higher degrees from universities. There is also a need for employees to be trained in particular specialized topics, and organizations typically offer many training options of varying types and durations for the benefit of its employees. Various aspects of training are discussed in Section 4.1.

- **Knowledge management initiatives**: A significant risk in many of the organizations that participated in the Helix interviews was the imminent loss of senior system engineers and their vast experiences. Many organizations have established initiatives to capture those experiences in various ways, and to store them in a readily accessible manner as when required.

- **Leadership development initiatives:** The most senior technical position that a systems engineer can achieve in an organization is that of a chief systems engineer or equivalent. Organizations tend to identify high-potential employees from its pool of junior and mid-level systems engineers, and offer them initiatives to enhance their leadership proficiencies in addition to technical proficiencies, thus enabling those systems engineers to develop in to future chief systems engineers and other senior systems engineering positions.

- **Rewards and recognition initiatives:** As a way to motivate, encourage, and appreciate the achievements of its systems engineers, organizations establish various rewards and recognition initiatives specifically for systems engineers in addition to its employees at large.

4.3.3 Phases of Organizational Initiatives

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

- **Identifying the need:** The first step in any organizational initiative is to clearly articulate the need for one, or define the problem that needs to be solved. While there are many types of initiatives that an organization could potentially establish, it is imperative for an organization to understand why a particular initiative is required.
- **Establishing the initiative**: Once the need is recognized and the type of initiative is identified, the organization must then establish the initiative by setting up the required policies, guidance, personnel to run / manage the initiative, criteria for selecting beneficiaries, and the required infrastructure.
- **Deploying the initiative:** There are a number of activities to be done once the organization has established an initiative:
  - *Promoting:* In 90% of the organizations that participated in Helix interviews, there were initiatives that at least one Helix interviewee was not aware of. The organization must take an effort to let its employees know of any initiative that they can benefit from. Newer employees who go through some sort of an orientation, tend to be more aware of initiatives that they can immediately benefit from. But those employees who have spent many years in the organization are not very aware of the initiatives that are available to them.
  - *Enabling*: When an employee is interested in a particular initiative and is eligible, the organization must enable the employee to benefit from that initiative. Experiences shared by Helix participants indicate that there are situations when they are unable to take advantage of an organizational initiative since they could not take time off their regular work to participate in a training initiative, or that some procedures diminished the effectiveness of the initiative.
- **Responding to outcomes of initiatives**: When an employee participates and benefits from an initiative, typically, there are new skills or knowledge that are acquired, and the employee could recommend improvements based on this. For example, if an employee receives education or training on systems engineering processes, and if the organization does not support modification of existing systems engineering processes, it defeats the purpose of the education.
- **Evaluating the initiative:** The most critical aspect of the success of an initiative is to evaluate it periodically, and to then update, reform, stop, or restart an organizational initiative. A critical evaluation could also reveal enablers and inhibitors for the initiatives. Helix interviews indicated evidence of many situations:
  - Initiatives no longer address the need for which they were established.
  - The need for which an initiative was established is no longer valid.
  - There are more trainers than trainees.
  - Employees are not motivated.
  - The evaluation of some initiatives makes it appear more successful than it really is.
  - The procedures and policies for an initiative could be burdensome.

- There is a need to restart an initiative that used to be very effective but was stopped due to many reasons, including budget cuts.
- The duration of a training course may be altered.
- The target beneficiaries for an initiative need to be redefined.

4.3.4 Critical Factors for Success with Organizational Initiatives

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

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

4.4 Conclusions

The findings presented in Section 4 provide an overview of the insights gleaned from additional data analyses to fill gaps in the previous iteration of *Atlas*, version 0.5. The patterns and characterizations presented above reflect the nature of the data. However, the reason the Helix team is interested in training, personal development initiatives, and organizational development initiatives is ultimately due to the fact that each of these are expected to have an impact on a systems engineer's proficiency, and ultimately effectiveness. Where possible, the team has highlighted insights on common patterns regarding proficiency; for example, training courses have the greatest and longest-lasting impact on proficiency when learnings from the training are applied in practice during or soon after the training event. Going forward to *Atlas* 1.0, the team will work focus implementation efforts and additional data collection on further clarifying the specific ways that these and other elements impact proficiency.

5 FUTURE DIRECTIONS

During the remainder of 2016, Helix will largely focus on creating and validating *Atlas 1.0*. There are many tasks that are planned for 2016 that would support development of *Atlas 1.0* and other research extensions. The major directions are:

- **Validating *Atlas***: Being developed using a grounded theory-based approach, *Atlas* is reflective of the data that has informed the building of the theory. Validity of the elements of *Atlas*, particularly of the proficiency model, and the career paths of systems engineers outside the population of the current Helix interviewees is to be established. To support this, validation will be an explicit objective of upcoming Helix site visits.
- **Expanding the Industry Sectors and System Types**: In December 2015, Helix published a report expanding *Atlas* beyond the DoD and DIB into other industry sectors in the US and included discussion and relevance to related, non-systems engineering professionals such as classic engineers and program managers. The Helix team will continue to look for key areas where elements of *Atlas* may differ between these groups as well as areas that are universally applicable.
- **Exploring Relationships Between *Atlas* Elements.** The team is aware of interplay between many of the different elements of *Atlas*. The clearest relationships are highlighted in *Atlas* today, as reflected in Figure 1. But there are many other subtle interplays between these elements. For example, there is overlap between training and education and many of the personal initiatives that systems engineers pursue are by taking advantage of optional organizationally sponsored training or educational opportunities. Mentors not only provide input on jobs but also on career paths, which include recommendations on training and education as well. This is only one example of the interplay between elements, but highlights the complexities and the multiple relationships between elements.
- **Strengthening the Dynamic View of *Atlas***: Career paths of systems engineers capture the dynamic aspect of *Atlas*. This will be further explored in 2016 not only to study the dynamic aspects leading up to the current level of proficiencies of individual systems engineers, but being able to predict future changes in proficiency levels as a result of specific forces. Appropriate modeling tools and simulation techniques will be explored.
- **Supporting Independent Atlas Deployment:** *Atlas 1.0* will not only present a more mature version of the theory of effective systems engineers, it will also provide several of the artifacts needed by individuals and organizations to independently use *Atlas* for personal and workforce development.
- **Engaging on a Larger-Scale:** Until now, organizational participation in Helix has been on a relatively modest scale in terms of the systems engineers who participated in Helix interviews, compared to the entire systems engineering population within an organization. If presented with the opportunity, Helix anticipates a larger-scale engagement of a systems engineering organization in Helix where a significant portion of the systems engineering population will be engaged; many of the individuals will not be systems engineers themselves, but interact with systems engineers; and a deeper understanding of the organizational characteristics will be obtained. The results of this larger-scale engagement may not readily transfer outside of that organization, but the exercise is expected to validate the anticipated value of *Atlas* to individuals and organizations. In addition, this may lead to patterns – at least within individual organizations

– of feasible, preferred, or possibly infeasible career paths. In other words, by understanding the proficiencies of the workforce, and tracking this and the career paths over time, it may be possible to identify when individuals are on appropriate or misaligned trajectories for particular positions.

- **Community Building:** Helix workshops will continue to be a forum for various Helix stakeholders to review the progress and plans of Helix; for organizations who have already participated in Helix to share the experiences; for other organizations to share their expectations; and for the systems engineering community art large to set the expectations for Helix that would be most valuable.

The 1.0 version of *Atlas* is planned for late 2016.

References

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

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

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

BKCASE Editorial Board. 2015. *The Guide to the Systems Engineering Body of Knowledge (SEBoK), v. 1.4*. R.D. Adcock (EIC). Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed 14 November 2014. www.sebokwiki.org.

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.

Bourque, L.B. 2004. "Coding." In *The Sage Encyclopedia of Social Science Research Methods*, edited by M.S. Lewis-Beck, A. Bryman, and T.F. Liao, v. 1, 132-136. Thousand Oaks, CA: Sage.

Brinkerhoff, R.O. and M.U. Montesino. 1995. "Partnerships for Training Transfer: Lessons from a Corporate Study." *Human Resource Development Quarterly*. 6(3): 263-274.

Burke, L.A. and H.M Hutchins. 2008. "Training Transfer: An Integrative Literature Review." *Human Resource Development Review*. 6(3): 263-296.

Business Dictionary. 2016. "Training."BusinessDictionary.com. http://www.businessdictionary.com/definition/training.html#ixzz44DpaGKr6

Clarke, N. 2002. "Job/work environment factors influencing training effectiveness within a human service agency: Some indicative support for Baldwin and Fords' transfer climate construct." *International Journal of Training and Development*. 6(3): 146-162.

DAU. 2016. "Glossary of Defense Acquisition Acronyms and Terms: training." Ft. Belvoir, VA: Defense Acquisition University (DAU).

Davidz, H.L. 2006. *Enabling Systems thinking to Accelerate the Development of Senior Systems Engineers*. Dissertation, Doctor of Philosophy in Engineering Systems. Massachusetts Institute of Technology.

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

Dedoose. 2016. "Dedoose User Guide." Accessed April 20, 2016. Available at: http://www.dedoose.com/userguide/.

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.

DoD. 2015. *Operation of the Defense Acquisition System*, DoD Instruction 5000.02. Arlington, VA: US Department of Defense.

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.

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

GAO. 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.

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

Goulding, C. 2002. *Grounded Theory: A Practical Guide for Management, Business and Market Researchers*. Thousand Oaks, CA: Sage.

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

IBM. 2016. “Rational DOORS.” Accessed April 19, 2016. Available at http://www-03.ibm.com/software/products/en/ratidoor.

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. 2015. “Certification.” Website of the International Council on Systems Engineering (INCOSE). Accessed April 29, 2015. Available at www.incose.org/certification.

INCOSE. 2015. *Systems Engineering Handbook*. San Diego, CA, USA: International Council on Systems Engineering.

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

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

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

Lim, D.H. and M.L. Morris. 2006. "Influence of trainee characteristics, instructional satisfaction, and organizational climate on perceived learning and training transfer." *Human Resource Development Quarterly*. 17(1): 85-115.

Lockyer, S. 2004. "Coding Qualitative Data." In *The Sage Encyclopedia of Social Science Research Methods*, Edited by M.S. Lewis-Beck, A. Bryman, and T.F. Liao, v. 1, 137-138. Thousand Oaks, CA: Sage.

MindMapper. 2016. "Harness the Power of Visual Mapper, Planner, and Dashboard." SimTech Systems, Inc. Accessed February 19, 2016. Available at http://mindmapper.com/.

QSR International. 2016. "NVIVO Products." Accessed 20 April 2016. Available at http://www.qsrinternational.com/product.

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, and D. Henry (eds.). 2015. *Graduate Reference Curriculum for Systems Engineering (GRCSE®) v.1.1*. Hoboken, NJ, USA: Trustees of the Stevens Institute of Technology.

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

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

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

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

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

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

SHRM. 2015. "HR Terms: Career Path." Alexandria, Virginia: Society for Human Research Management website. Last updated 2015. Accessed May 27, 2015. Available at: http://www.shrm.org/templatestools/glossaries/hrterms/pages/c.aspx.

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.

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.

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.

ACRONYMS & ABBREVIATIONS

DASD(SE)	U.S. Deputy Assistant Secretary of Defense for Systems Engineering
DIB	Defense Industrial Base (that supports DoD)
DoD	U.S. Department of Defense
GRCSE	Graduate Reference Curriculum for Systems Engineering
INCOSE	International Council on Systems Engineering
IT	Information Technology
NDIA-SED	National Defense Industrial Association – Systems Engineering Division
SE	Systems Engineering
SERC	Systems Engineering Research Center
SEBoK	*Guide to the Systems Engineering Body of Knowledge*
SME	Subject Matter Expert
SPRDE	Systems Planning, Research, Development, and Engineering
UARC	University-Affiliated Research Center
PM	Project (or Program) Manager

APPENDIX A: SELF-ASSESSMENT TOOLS: PROFICIENCY AND CAREER PATHS

This appendix provides a copy of the tools created by the Helix team for self-assessment of an individual's career path, including their proficiency. For ease of printing, these tools begin on the next page. Additional detail on the career path method can be found in Section 3. If a reader has any questions or concerns about the use of these tools, they are encouraged to contact the Helix team at helix@stevens.edu. These tools are expected to evolve significantly during 2016 in the lead up to the release of *Atlas 1.0.*

Instructions for Completing a Proficiency Self-Assessment

Overview

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

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

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

Self-Assessment

In order to perform a self-assessment, individuals are asked to review the definitions of the proficiency areas above and the categories in Table 1. Additional detail can be found in the full report on *Atlas* 0.5, found at http://cdn.sercuarc.org/wp-content/uploads/2014/05/Helix-Report-Atlas-0.5-December-2015.pdf. Then use the template to generate a "0 to 10" initial assessment of your current proficiency in each Area, with "0" meaning you have no skill in the area and 10 meaning your skills are the top within your experiences. Consider the following guidelines:

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

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

Area	Category	Topic
1. Math / Science / General Engineering	1.1. Natural Science Foundations	
	1.2. Engineering Fundamentals	
	1.3. Probability & Statistics	
	1.4. Calculus & Analytical Geometry	
	1.5. Computing Fundamentals	
2. Systems' Domain & Operational Context	2.1. Relevant Domains	
	2.2. Relevant Technologies & Systems	
	2.3. Relevant Disciplines	
	2.4. Familiarity with System's Concept of Operations (ConOps)	
3. Systems Engineering Discipline	3.1. Lifecycle	3.1.1 Lifecycle Models; 3.1.2 Concept Definition; 3.1.3 System Definition; 3.1.4 System Realization; 3.1.5 System Deployment & Use; 3.1.6 Product & Service Life Management
	3.2. Systems Engineering Management	3.2.1 Planning; Risk Management; 3.2.2 Configuration Management; 3.2.3 Assessment & Control; 3.2.4 Quality Management
	3.3. SE Methods, Processes, & Tools	3.3.1 Balance & Optimization; 3.3.2 Modeling & Simulation; 3.3.3 Development Process; 3.3.4 Systems Engineering Tools
	3.4. System Complexity	
4. Systems Engineering Mindset	4.1. Big-Picture Thinking	
	4.2. Paradoxical Mindset	4.2.1 Big-Picture Thinking and Attention to Detail; 4.2.1 Strategic and Tactical; 4.2.1 Analytic and Synthetic; 4.2.1 Courageous and Humble; 4.2.1 Methodical and Creative
	4.3. Flexible Comfort Zone	
	4.4. Abstraction	
	4.5. Foresight & Vision	
5. Interpersonal Skills	5.1. Communication	5.1.1 Audience; 5.1.2 Content; 5.1.3 Mode
	5.2. Listening & Comprehension	
	5.3. Working in a Team	
	5.4. Influence, Persuasion & Negotiation	
	5.5. Building a Social Network	
6. Technical Leadership	6.1. Building & Orchestrating a Diverse Team	
	6.2. Balanced Decision Making & Rational Risk Taking	
	6.3. Managing Stakeholders and their Needs	
	6.4. Conflict Resolution & Barrier Breaking	
	6.5. Business & Project Management Skills	

Position Assessment

Position: _______________

- Probability and Statistics
- Calculus & Analytical Geometry
- Natural Science Foundations
- Engineering Fundamentals
- Computing Fundamentals

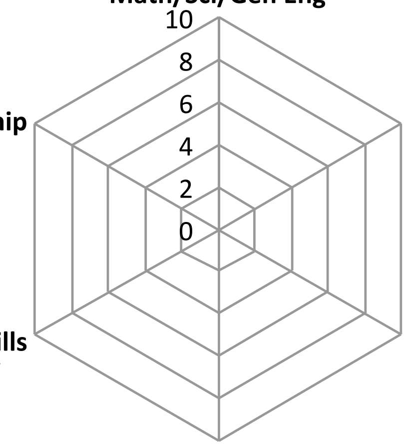
Notes:

- Relevant Domains
- Relevant Technologies and Systems
- Relevant Disciplines
- Familiarity with System's Concept of Operations

Notes:

- Building & Orchestrating a Diverse Team
- Balanced Decisions Making & Risk Taking
- Managing Stakeholders and Their Needs
- Conflict Resolution & Barrier Breaking
- Business & Project Management Skills

Notes:



- Lifecycle
- SE Management
- SE Methods, Processes, & Tools
- System Complexity

Notes:

- Communication
- Listening & Comprehension
- Working in a Team
- Influence, Persuasion, & Negotiation
- Building a Social Network

Notes:

Proficiency of a Systems Engineer

- Big Picture Thinking
- Paradoxical Mindset
- Flexible Comfort Zone
- Quick Learning & Abstraction
- Inquisitive & Self-Driven
- Foresight & Vision

Notes:

Instructions for Completing a Career Path Assessment

Overview

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

Experiences

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

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

- **Relevance:** A 'relevant' position is one that enables a systems engineer to develop the proficiencies critical to systems engineering. Determine a starting point for relevant experiences; this will become the first position (P1) of the career path. Fill in the title and the year(s) for the position(s).
- **Organizations:** Fill out the name of the organization for each position. This will help to show any transition or variation between organizations.
- **Roles:** A role is a collection of related systems engineering activities. Roles were identified based on the activities consistently performed by systems engineers. There are 16 roles identified in *Atlas*, as described in Table 1, below. For each position, review your activities and responsibilities and write down *all* roles played during that position.
- **Lifecycle Phases:** Generic systems engineering lifecycle phases considered in *Atlas* are based on the lifecycle phases in the *Guide to the Systems Engineering Body of Knowledge (SEBoK*), as explained on page 5. (BKCASE Authors 2015) For each position, fill in the area(s) of the lifecycle you worked on.
- **Key Milestones.** Note any key changes in types of positions under key milestones. For example, first systems engineering role, first chief systems engineer role, first supervisory position, etc. would all be indicators of change or growth over career.

Education and Training

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

Other

Your organization may ask you to add other information or you may wish to capture additional information. Examples include key mentoring experiences, participation in professional societies, publications, etc. to your career path.

Role (Abbreviation)	Description
Requirements Owner (RO)	Individual responsible for translating customer requirements to system or sub-system requirements or developing the *functional* architecture.
System Designer (SD)	Individual responsible for owning or architecting the system; common titles may includes CSE, system architect.
System Analyst (SA)	Individual who provides modeling or analysis support to system development activities and helps to ensure that the system as designed should meet specification.
V&V Engineer (VV)	Individual who plans and conducts verification and validation activities, such as testing, demonstration, simulation, etc.
Logistics/ Operations Engineer (LO)	Individual who performs the 'back end' of the SE life cycle, who may operate the system, provide support during operation, provide guidance on maintenance, or help with disposal.
Glue (GL)	Individual who is responsible for a holistic perspective on the system; from Sheard (1996), this may be the "technical conscience" or "seeker of issues that fall 'in the cracks'", particularly someone who is concerned with interfaces.
Customer Interface (CI)	Individual who is responsible for coordinating with the customer, particularly for ensuring that the customer understands technical detail and that a customer's desires are, in turn, communicated to the technical team.
Technical Manager (TM)	Individual who is responsible for controlling cost, schedule, and resources for the technical aspects of a system; often someone who works in coordination with an overall project or program manager.
Information Manager (IM)	Individual responsible for the flow of information in a system development activity; specific activities may include configuration management, data management, or metrics.
Process Engineer (PE)	Individual responsible for the SE process as a whole, who likely has ties into the business directly.
Coordinator (CO)	Individual responsible for coordinating amongst a broad set of individuals or groups who help to resolve systems issues; key associated skills would include negotiation, mediation, and communication.
Systems Engineering Evangelist (EV)	A common role that SErs reported play is that of "evangelist" – an individual who has to promote the value of SE to individuals outside of the SE community. This may be to project managers, other engineers, or management. Individuals who played this role consistently stated that they had to play this role in order to effective – sometimes to be allowed to play other SE roles. This role was added based on the Helix analysis and was suggested as a potential addition in (Sheard 2000).

Detailed Designer (DD)	Individual who provides technical designs that match the system architecture; an individual contributor in any engineering discipline who provides part of the design for the overall system. This is not a SE role, so is not included in Sheard's roles (1996). However, as many SErs start in specialty design, and interviewees consistently stated that these roles were valuable to their growth into SE roles.
Organizational/ Functional Manager (MG)	Individual responsible for the personnel management of SErs or other technical personnel in a business – not a project or program – setting. While this is not a "systems engineering" role, it does provide opportunities for individuals to build non-technical skills such as leadership and communication and is, therefore, included in the analysis for SErs.
Instructor/ Teacher (IN)	An individual responsible for providing or overseeing instruction of SE discipline, practices, processes, etc. While not a "systems engineering' role, per se, an individual who provides training or education must have some level of mastery of the subject matter and has multiple opportunities to improve skills such as communication.
Program/Project Manager (PM)	An individual who performs program or project management activities. As defined by PMI, project management is the application of knowledge, skills, tools, and techniques to project activities to meet the project requirements. (PMI 2012) Program management is the application of knowledge, skills, tools, and techniques to a program to meet the program requirements and to obtain benefits and control not available by managing project individually. (PMI 2012) A program or project manager is not directly responsible for the technical content of a program, but works closely with technical experts and other systems engineers. Therefore, this is considered a relevant role for this analysis.

Systems Engineering Lifecycle

- **Concept Definition** - A set of core technical activities of SE in which the problem space and the needs of the stakeholders are closely examined. This consists of analysis of the problem space, business or mission analysis, and the definition of stakeholder needs for required services within it.
- **System Definition** - A set of core technical activities of SE, including the activities that are completed primarily in the front-end portion of the system design. This consists of the definition of system requirements, the design of one or more logical and physical architectures, and analysis and selection between possible solution options.
- **System Realization** - The activities required to build a system, integrate disparate system elements, and ensure that a system both meets the needs of stakeholders and aligns with the requirements identified in the system definition stage. This includes integration, verification, and validation (IV&V).
- **System Deployment and Use** - A set of core technical activities of SE to ensure that the developed system is operationally acceptable and that the responsibility for the effective, efficient, and safe operations of the system is transferred to the owner. Considerations for deployment and use must be included throughout the system life cycle. Activities within this stage include deployment, operation, maintenance, and logistics.
- **Product and Service Life Management** - Deals with the overall life cycle planning and support of a system. The life of a product or service spans a considerably longer period of time than the time required to design and develop the system. This stage includes service life extension, updates, upgrades, and modernization, and disposal and retirement. The

organizations in the current sample are primarily concentrated on new development, so this is a very under-represented aspect of the life cycle.

- In addition to these life cycle phases, the SEBoK includes orthogonal activities of systems engineers, **Systems Engineering Management**, defined as managing the resources and assets allocated to perform SE activities. Activities include planning, assessment and control, risk management, measurement, decision management, configuration management, information management, and quality management. These activities can occur at any point in the systems engineering lifecycle.

Career Path Template