LESSONS FROM SYSTEMS ENGINEERING FAILURES: DETERMINING WHY SYSTEMS FAIL, THE STATE OF SYSTEMS ENGINEERING EDUCATION, AND BUILDING AN EVIDENCE-BASED NETWORK TO HELP SYSTEMS ENGINEERS IDENTIFY AND FIX PROBLEMS ON COMPLEX PROJECTS

by

Diane Constance Aloisio

A Dissertation

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Doctor of Philosophy

School of Aeronautics & Astronautics

West Lafayette, Indiana

December 2018

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Karen Marais, Chair

School of Aeronautics and Astronautics

Dr. Edward Berger

School of Engineering Education and School of Mechanical Engineering

Dr. Steven Landry

School of Industrial Engineering and School of Aeronautics and Astronautics

Dr. Roshanak Nateghi

School of Industrial Engineering

Approved by:

Dr. Weinong Chen

Head of the Graduate Program

I would like to thank my family, who has supported me during my time in graduate school. Thank you mom and dad, who not only helped me apply to graduate schools but also helped me apply to jobs during the last year. Thank you for helping us move here, coming to visit, and always answering my needy phone calls. Thank you dad for dragging me to airplane museums as a child and inspiring me to be an aerospace engineer. Thank you mom for being my emotional rock and keeping me sane.

To my husband, Matthew, thank you for moving across the country with me so I could follow my dreams. Thank you for encouraging me to continue when graduate school started to really get stressful. Without you here with me I am not sure I would have finished and I definitely would not have eaten many home-cooked, healthy meals.

To my friends, those who I knew before graduate school and supported me moving away and those who I met while in graduate school, thank you for making each day a little brighter.

I would like to thank all the people who have helped make this work possible over the years: my statistics consultants, Hanxi Sun and Wenbin Yu, and the Statistics Consulting Service at Purdue, who helped make the statistics analysis of this document so robust; the undergraduate researchers: Weronika Jusczak, Krysta Rolle, Kimberly Rink, and Derek Biega, who not only helped me with research work but also helped me cultivate my mentorship skills; the staff at the Center for Instructional Excellence: Dave Nelson, Craig Zywicki, and Emily Bonem, without whom I would not have any survey data; Zach Yater, whose programming skills made the cause-recommendation network interactive and allowed us to put it on the internet; and Dr. Colin Gray who went through the network with me, identified its usability problems, and gave us invaluable ideas on how to fix those problems.

Thank you to the members of the VRSS research group, past and present. Despite all of us suffering through graduate school, we have cultivated an environment where even though our research topics are different we still provide helpful advice to each other. I am pleased to observe that we are now all "the favorite" as a unit.

Last and most importantly, I would like to thank Dr. Karen Marais, my adviser. Thank you for taking a chance on me as a new graduate student, for challenging me to always do my best work, and for being there on those late Sunday nights before a paper was due. Because of your guidance, I have become a much better writer, mentor, and presenter, and I have become a well-rounded professional. I could always rely on you for support, whether it was during a difficult meeting, when I was applying to scholarships, or during my job search. Our relationship has improved during my time at Purdue and I hope that your incredibly hard work continues to be recognized.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. 1350589.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Author: Aloisio, Diane, C. PhD
Institution: Purdue University
Degree Received: December 2018
Title: Lessons from Systems Engineering Failures: Determining Why Systems Fail, the State of Systems Engineering Education, and Building an Evidence-Based Network to Help Systems Engineers Identify and Fix Problems on Complex Projects
Committee Chair: Karen Marais

As the complexity of systems increases, so does what can go wrong with them. For example, the United States Air Force selected McDonnell Douglas' design for the F-15 Eagle fighter aircraft in 1967 and the aircraft's first test flight was in 1972, 5 years later. In contrast, the US military selected Lockheed Martin as one of two companies to develop the F-35 Lightning II in 1997 and its first flight was in 2006, 9 years later, and the first production aircraft had its first flight in 2011, 14 years after the selection. This complex program's problems have been well-documented by the U.S. Government Accountability Office (GAO) and have contributed to the project's long lead time and skyrocketing budget. GAO reports on other military projects reveal that problems the F-35 project has experienced are shared among all of these projects. In this dissertation I posit that similar problems plague all complex systems engineering projects and that a combination of these problems may lead to negative consequences, such as budget and schedule exceedances, quality concerns, not achieving mission objectives, as well as accidents resulting in loss of human life.

Accidents, or unexpected events resulting in loss, have been well-studied over time and we currently have sophisticated theories that help explain how they occur. The leading theory is that most accidents are a result of an accumulation of "mundane" errors at an organization, and that these errors are similar across industries. However, these mundane errors, such as failing to follow procedures and poorly training personnel, occur in all companies, such as companies that design and manufacture military aircraft. My theory is that these mundane errors accumulate in all organizations and result in many different kinds of systems engineering failures, including failures traditionally referred to as "accidents" that result in loss of life, as well as other types of failures which I refer to as "project failures".

What can be learned from these systems engineering failures? In this dissertation, I begin by mining publicly-available reports to determine whether seemingly dissimilar failures, accidents and project failures, share common causes. I then explain the similarities and dissimilarities between these causes and provide examples from the failures I studied. To help provide systems engineers with actionable advice on these common causes, I describe how I linked the causes to recommendations from accident reports in a cause-recommendation network. I then discuss the results of interviews I held with systems engineers to determine whether the problems I identified in past failures occur in similar ways to the problems they have encountered on their projects. I also discuss the criticisms these systems engineers have about systems engineering education based on the tasks their newly-hired systems engineers struggle with. I explain how I used what I learned about problems in systems engineering that lead to failures to develop survey questions designed to gauge whether systems engineering education at Purdue prepares students to identify and fix these problems. Then, to help systems engineers learn from the data I collected and solve the problems they encounter on their projects, I describe how I built an interactive, web-based tool that presents expert advice on systems engineering failures. I finally explain the results from feedback I received from experts and novices in systems engineering to determine whether this tool could be useful for engineers in this context.

CHAPTER 1. INTRODUCTION AND BACKGROUND

1.1 Motivation

Engineering projects are infrequently completed on time, within budget, and with the required features and functions, and their problems are growing as projects become more complex. For example, the United States Air Force selected McDonnell Douglas' design for the F-15 Eagle fighter aircraft in 1967 and the aircraft's first flight was in 1972, 5 years later [The Boeing Company, 2018]. In contrast, the US military selected Lockheed Martin as one of two companies to develop the F-35 Lightning II in 1997 and its first flight was in 2006, 9 years later, and the first production aircraft had its first flight in 2011, 14 years later [Lockheed Martin, 2018]. More specifically, of 72 major United States defense programs in progress in 2008, only eleven of them were on time, on budget, and met performance criteria [Charette, 2008]. The problems for U.S. aerospace and defense programs have only worsened since then: total cost overruns "have risen from 28 percent to 48 percent, from 2007 through 2015" [Lineberger & Hussain, 2016]. In a recent assessment of U.S. Defense Acquisitions, the U.S. Government Accountability Office (GAO) suggested that many current programs are vulnerable to "cost growth or schedule delays" [GAO, 2017]. The consumer goods sector also has many failures, such as the Xbox 360 "Red Rings of Death" or the Ford Explorer rollover problems [Takahashi, 2008; Bradsher, 2000].

We propose that these failures occur in systems engineering due to three major factors: systems engineering practitioners are not learning from past failures to improve their systems engineering efforts, there are gaps in current systems engineering education that lead to new systems engineers being unprepared for problems they encounter on projects, and there is little actionable guidance for systems engineers experiencing problems.

1.2 Approach

How have systems engineers attempted to learn from past failures so far? Previous studies on project failures have identified problems in systems engineering, and other studies have identified problems in project management, but neither the research community nor practitioners have been able to identify fully and prevent the causes of project failures. Shore [2008] studied a variety of project failures and identified “systematic biases” throughout these cases, such as “overconfidence” and “conservatism”. Williams et al. [2012] studied eight cases of how project assessments identified early warning signs of impending project failure. They provide descriptive cases of warning signs at various stages in the project lifecycle (e.g. “lack of a culture of openness and good communication between actors” during the early stages of a project). Keil & Mähring [2010] performed an in-depth study of two IT project failures: the Eurobank deposit system and the California DMV database. They analyzed these failures using an “escalation” framework, which describes three distinct phases project failures experience: drifting, treating symptoms, and rationalizing continuation. Nutt [2002] identifies poor decisions in a wide range of industries in 15 detailed case studies. For example, he describes how “ambiguous directions” led Disney to build EuroDisney to realize “Walt’s Dream”, rather than basing its decisions on market demand and profitability.

At the other extreme, Newman [2001] analyzed 50 space failures from a high-level systems engineering perspective and discussed broad categories into which many of the causes of these failures fell (e.g., Design, Manufacturing, or Human Error). Konstandinidou et al. [2011] studied over 1,000 incidents in the Greek petrochemical industry and identified causal factors, such as “inadequate procedures” and “lack of communication” that contributed to these incidents. They found that causal factors such as human factors (e.g. “errors of omission”) contributed more to injury and workplace absences, and organizational factors (e.g. “inadequate procedures”, “inadequate training”) contributed more to material damage. These types of analyses aim to achieve formal statistical significance or at least strength in numbers, but may sacrifice the depth of the case study approach.

Accident causation has also been extensively studied and researchers and many practitioners now have a sophisticated understanding of why accidents occur. Industries like commercial aviation have been able to translate these findings into remarkable safety records, and continue to seek new ways to further improve safety. One of the key findings is that the root causes of most accidents lie in organizations and their people rather than in the particular technologies being used, and, that the same root causes can be found in accidents that appear different on the surface [Saleh et al., 2010; Leveson, 2004]. For example, both the Three Mile Island partial nuclear meltdown and the Alaska Airlines Flight 261 aircraft crash resulted from repairs that were poorly done because the components requiring repair were not designed with human accessibility in mind [Kemeny et al., 1979; NTSB, 2000].

In this dissertation, we synthesize these approaches by studying many systems engineering failures in-depth to give our study the “strength in numbers” benefit while also retaining the specific criticisms from each failure. Our approach is novel in that we conflate accident and project failure causation to create a database of causes that apply to both.

Perhaps systems engineering failures are more common within the last few decades because the complexity of the systems we build is increasing. As a result, the demand for systems engineers is also increasing [Hutchison et al., 2016; SERC, 2013; Chaput & Mark, 2013]. 23% of all engineers in the U.S. are over the age of 55, which means there may be a labor shortage in the near future as these engineers begin to retire [Wright, 2014]. Retiring systems engineers, specifically, are a major concern in the defense industry [SERC, 2013; Charette, 2008] as well as at NASA [Bagg et al., 2003]. One obvious solution is to train more undergraduates in systems engineering skills. However, there is a pervasive belief that successful systems engineers can only be made through experience [e.g. Armstrong & Wade, 2015; Squires et al., 2011; Davidz et al., 2005]. This belief may partially be due to the previous generation of systems engineers not receiving much systems engineering-specific training in their university engineering education, as noted by Armstrong & Wade [2015] in their interview-guided study on how systems engineers develop their expertise. Additionally, many systems engineers have an integrative role, “requiring a deeper understanding of a wide range of areas than provided by a focused education” [Ross et al., 2014]. Anecdotally, many university faculty agree that successful systems engineers can only be made through experience, as evidenced in part by the relatively few programs in systems engineering, especially at the undergraduate level[1]. Of the top twenty engineering programs in the United States, only four list systems engineering as a major for undergraduates: University of Illinois ("Systems Engineering and Design"), Texas A&M University ("Industrial and Systems Engineering") University of Southern California ("Industrial and Systems Engineering"), and University of Pennsylvania ("System Science and Engineering") [U.S. News, 2018; University of Illinois, 2018; Texas A&M University, 2018; University of Southern California, 2018; University of Pennsylvania, 2018]). As Adcock et al. [2015] note: "current undergraduate engineering education lacks systems education in key areas". In aerospace engineering in particular (many graduates of which are hired to the defense industry), "teaching SE [Systems Engineering] is not a significant part of our undergraduate aerospace engineering design course objectives" [Chaput, 2010]. Currently, most systems engineers start out as engineers in more traditional engineering areas, like structures or flight testing. Despite the interdisciplinary and integrative nature of many systems engineering efforts in practice, "if SE is taught at all, it is taught as a separate subject" [Chaput, 2016]. As a result, newly-graduated engineers from these traditional engineering disciplines often do not have the necessary systems engineering skills to help projects succeed and "need to be grown via in-house training or experience" [Adcock et al., 2015]. For example, NASA developed the Systems Engineering Leadership Development Program (SELDP) to provide "development activities, training, and education" to more quickly cultivate systems engineers [Ryschkewitch et al., 2009].

Universities have responded to the growing market demand for systems engineers in a range of ways, from adding or further emphasizing elements of systems engineering to existing courses (e.g., capstone design courses; see Chaput [2016]), to creating entire programs in systems engineering (e.g., Stevens Institute of Technology). How effective are these efforts, how can they be improved, and can we identify a set of best practices in doing such training [cf. Squires et al., 2011]? Here, we address, in part, the first question. Our approach is based on assessing how well

[1] A brief note on terminology is appropriate here. While there are many graduate engineering programs that address the problems posed by complex engineering systems, these programs tend to focus on the science of engineering systems, and generally do not claim to produce systems engineers, rather, they produce graduates skilled in aspects of system development and operation.

students can identify and address problems that have resulted in previous system development or operation failures.

There are several standardized tests intended to gauge critical thinking ability, such as the Watson-Glaser Critical Thinking Appraisal, the Cornell Critical Thinking Test, and the California Critical Thinking Skills Test [Jacobs, 1999], but these tests do not gauge systems engineering-specific abilities. Researchers in engineering fields, such as chemical engineering, have created and deployed tests called "knowledge-base evaluation" designed to evaluate students' mastery of primary foundational topics in these fields [Farand & Tavares, 2017]. These researchers are trying to address a problem in chemical engineering education: their students are able to resolve complex problems but have difficulty explaining the concepts underlying their calculations, such as basic fluid mechanics and heat transfer concepts. The researchers deployed their test to students taking a mandatory fourth-year course on a computer in an exam scenario. They collected and analyzed data from 4 years of testing, and now use their tool to not only assess student ability to learn key concepts in chemical engineering, but also to collect feedback on courses and improve their educational program.

Our approach to testing student systems engineering ability is different to that of Farand & Tavares [2017] in that we based our survey questions on the problems we identified while studying systems engineering failures, as opposed to basing our survey questions on students' performance. The problems we identified in these systems engineering failures may indicate inadequacies in systems engineering education, so if we study how well students do on questions based on these problems, it may indicate where the biggest gaps in systems engineering education are.

Once well-trained systems engineers identify problems on their projects, what can they do about it? Guidance on preventing failures is often quite general, such as "put your best people on the project and resolve the root causes" [Keil & Mähring, 2010] or "top management needs to provide unambiguous reinforcing messages from time to time" [Chanda & Ray, 2015]. Such guidance is certainly valid (clearly one would not want to put one's worst people on an important project!), but it also tends to only address problems at the surface and not the underlying reasons for these problems (Why weren't the best people on the project? And where should the not-best people be placed?). So, guidance is usually either highly contextualized or very general.

One reason for this state of affairs is that information on project failure causation is generally difficult to acquire. Few organizations publicize specific details on project failures (for example, exactly how and why a valve failed, or why the budget was exceeded and by how much), and, unless failures involve extensive damage or financial loss, injury, or death (in other words, cases where investigations are legally or otherwise (e.g., congressionally) mandated), investigations are usually at the discretion of the organization that was directly involved in them (cf. OSHA, NTSB). In contrast, accidents are a type of failure that is less susceptible to information-unavailability and lack-of-specific-guidance problems. Organizations have to investigate accidents to some minimum level. Independent committees with significant resources and power often investigate large accidents and make the results of the investigation available to the public. These large investigations usually result in both specific recommendations as well as general guidance.

In the final part of this dissertation, we leverage conflating accident and project failure causation by linking the causes to recommendations in accident reports made by accident investigators to provide actionable guidance to systems engineers experiencing the same problems that may lead to project failures.

1.3 Thesis Outline

In Section 2, we use the lessons learned from accidents to help identify causes and preventive measures for other project failures. We use our findings to develop a cause-recommendation network that shows how causes tend to cluster, which recommendations are appropriate for which causes, and how the causes and recommendations are manifested in a range of industries.

Our test of systems engineering skills, described in Section 3, is inspired by the idea of foundational concepts. In our case, we base the foundational concepts on the errors that frequently lead to failures in complex engineered systems. This approach allows us to circumvent some of the potential “motherhood and apple pie” aspects of systems engineering (e.g., most students know that stakeholder needs should be considered during development—fully and appropriately doing so in practice is much more difficult).

In Section 4 we compile all of the information on failure causation in systems engineering to present an interactive solution aid that puts these causes and recommendations at an organization’s fingertips—providing an “instant-expertise” tool for anyone investigating failure causation and remediation. This tool bridges the gap between extremes of project management and systems engineering literature (guidance based on single case studies that is too specific to guidance based on massive studies of failures that is too general) by providing specific examples of causes and recommendations from failures we studied. Not only have we compiled causes and remediation measures from different failures across a variety of industries (such as the aerospace, nuclear, and coal mining industries), we have also extracted the salient details from these findings. Thus, an organization using our tool does not need to employ experts in nuclear reactor design or aircraft design to understand the fundamental ideas behind these findings and find our tool useful.

CHAPTER 2. WHY DO SYSTEMS FAIL?

In this section we explore why systems fail, beginning with reviewing accident research, then describing which failures we studied, how we extracted and coded findings from their reports. We then discuss how the causes of these failures are different between accidents and project failures and what reasons for those differences may be using illustrative examples. We build network graphs of the causes for accidents and project failures, then link causes and recommendations from accident reports to build a cause-recommendation network. Finally, we discuss the results of surveys we conducted with large-scale aerospace companies to determine whether the problems we identified in past systems engineering failures correspond to problems systems engineers in those organizations encounter.

We presented an earlier iteration of this work in Sorenson [2015], which summarizes accident literature more in a more in-depth way (including discussing a brief history of accident models), describes our failure analysis, how we modified and subsequently applied the STAMP model to each failure, and our coding process more thoroughly. Since publishing that document, we re-analyzed each failure and further refined the resulting codes, which we present in this section.

2.1 Review of Accident Research

A range of accident modeling techniques that help explain how accidents are caused is available. Accident investigation reports, and subsequent meta-analyses of these reports, have revealed that accidents across industries have similar causes despite occurring in different scenarios. This section provides a brief review of the literature; for a more extensive discussion see Saleh et al. [2010].

Theories and models on accident causation have become increasingly sophisticated, beginning from considering accidents as simple chains of human errors and physical failures to our current understanding that accidents result from a complex web of interactions, many of which are, or at least appear to be, locally and temporally rational. Man-made disasters theory is an early and influential articulation of this perspective [Turner, 1978]. It posits that accidents are not the result of chance events, but rather occur as a result of a build-up of errors and hazards over time. Man-made disasters theory helps explain why accidents occur even at organizations that have safety programs in place and claim to value safety. When members of the organization collectively follow the safety rules and procedures less well and less frequently or commit other mundane day-to-day errors, accidents may arise.

Human factors (ergonomics) and organizational factors studies have provided understanding of why people make errors. For example, people routinely violate procedures—because doing so often allows them to perform tasks more quickly and efficiently, sometimes at the cost of safety. James Reason's work, of which the Swiss cheese model is one of the best-know aspects, is an influential successor to Turner's work [Reason, 1990]. The Swiss cheese model views safety as being maintained by layers of defense, which develop and close holes over time as people follow or do not follow procedures, for example. When there are sufficient holes, or when holes remain in place for long enough, accidents can shoot through the layers of defense. Reason also posited that accidents can be traced back to problems on four levels: specific acts, preconditions, supervision, and organizational influences. Each higher level in the hierarchy drives the problems below it (i.e. the government drives the problems the regulators encounter, and the regulators drive the problems the company encounters. Based on these layers, Shappell and Wiegmann [2000] developed a taxonomy of accident causes and codified them in the Human Factors Analysis and Classification System (HFACS).

The view that system safety is a control problem that requires a systems perspective has emerged as the current leading theory. The control-theoretic perspective on system safety sees accidents as resulting from the absence or breach of defenses, be they technical or organizational, or from the violation of technical or organizational safety constraints [Rasmussen, 1997; Svedung & Rasmussen, 2002; Leveson, 2004; Saleh et al., 2010]. Absences and breaches of defenses and safety constraint violations can occur at any level of an organization. The Systems Theory Accident Modeling and Processes (STAMP) is one example of a model that considers accidents from a control system perspective. Shown in Figure 2.1, STAMP incorporates ideas from other models, such as Rasmussen's [1997] hierarchies, while showing examples of what problems can occur at different points in a product's lifecycle.

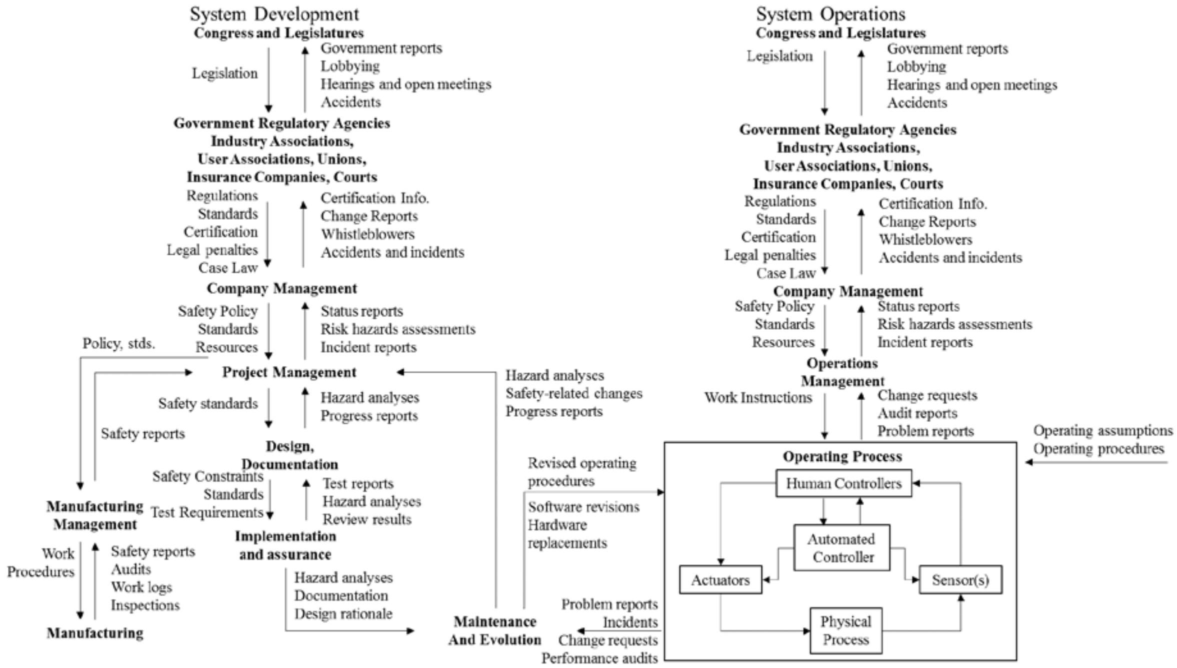


Figure 2.1: STAMP model, adapted from Leveson [2006]

Progress in accident theory and modelling is both informed by and also drives the growing recognition that accidents, though often differing in their details, share root causes, whether expressed as lurking pathogens in Swiss Cheese, layers or types of errors in HFACS, or control flaws in Rasmussen or Leveson’s work [e.g., Marais et al., 2006; Svedung & Rasmussen, 2002; Newman, 2001]. For example, the technicians working on the NOAA N-Prime Satellite committed a skill-based memory lapse error when they failed to notice that bolts holding the spacecraft to a working surface were missing, despite wiping the surface and not detecting interference from the bolts, resulting in the spacecraft toppling when they attempted to move the working surface [NASA, 2004]. After a Boeing 747 operated by China Airlines experienced a tailstrike incident (in which the tail of the aircraft struck the runway surface while the aircraft was taking off), personnel committed a rule-based mistake and did not follow maintenance procedures requiring them to remove the entire potentially damaged portion of the tail. The material eventually fatigued to the point of failure on flight 611 [ASC, 2002]. Both of these failures had problems with their organizational climates and communication: the NOAA N-Prime crew had an atypical mix of authority on the morning of the incident (the crew normally did not work together), which was not conducive to open discussion and shared responsibility, and the Boeing repair procedures and customer communications channels did not instruct the China Airlines crew on how to perform tailstrike repair correctly.

Here, then, we posit that, just as accidents share many causes, project failures more generally share causes with accidents in particular, and with project failures more generally. We explore this idea in the next section.

2.2 Cause Extraction and Comparison Methodology

This section describes the data set for our analysis, including the types of failures we studied and the nature of the reports on these failures. We then illustrate how we coded each finding from the failure reports and decomposed them into an easily comparable format that describes what occurred and who and what was involved.

2.2.1 Accident and Failure Selection

Our approach to understanding project failure causation is based on the ideas that (1) accidents share causes and (2) that project failures and accidents also share causes. We would also like to identify causes at a finer level of detail than provided by the general studies referred to in the introduction, but in a more generalizable form than that provided by individual case studies. Our focus is on systems engineering, so we exclude cases with causes that appear to be beyond the control of systems engineers (e.g., the Germanwings flight 9525 crash, which was deliberately caused by the co-pilot [BEA, 2016]).

Both accidents and project failures are “undesired and unplanned (but not necessarily unexpected) event[s] that result in (at least) a specified level of loss” [Leveson, 1995]. Here, we use the term “accident” to refer to those events that directly result in loss of life, injury, or damage to property. We use the term “project failures” for all other undesired project events, such as failure to achieve mission objectives, budget or schedule overruns, cancellations, or quality or performance issues. We identified the project failures we studied “in hindsight”, or after these projects had already met these criteria. It is more difficult to identify a project failure when it first starts to develop, as discussed in Williams et al. [2012] (which is also true for accidents).

There are relatively few detailed public investigations of project failures. We identified 33 cases with systems engineering-related causes, with sufficient detail, that span a range of industries, and that occurred relatively recently (from 1979 to 2015). In contrast, there is a plethora of detailed accident investigation reports. We considered 30 accidents from a wide range of industries. Table 2.1 shows our data set[2] and the references we read for each failure.

Table 2.1: Accidents and project failures

Failure Category	Industry	Case	Source(s)
Project Failure	Consumer product	Apple Newton MesssagePad	[Honan, 2013; Hormby, 2013]
		HD DVD	[Dilger, 2008]
		Google Glass	[Marks, 2014; Pachal, 2015]
		Zune	[Bylin, 2011; Enderle, 2011; Mokey, 2009; Warren, 2012; Wilcox, 2010]
		Windows Vista	[Dvorak, 2008; McCracken, 2009; Nelson, 2009; Smith, 2008]
		Xbox 360	[Takahashi, 2008; Wolverton & Takahashi, 2007; Yam, 2008]
		Ford Explorer	[Bradsher, 2000; Claybrook, 2000; Kumar, 2001; Whoriskey, 2010]
		Merck Vioxx Drug	[Topol, 2004]
		Boeing 787 Dreamliner	[Denning, 2013; Mouawad, 2014]
		Iridium Satellite Phone	[Cowing, 2000; Finkelstein & Sanford, 2000; Mcintosh, 1999; Mellow, 2004]
		Segway	[Hollmer, 2002; Pachal, 2015; Rivlin, 2003]
	Infrastructure project	FAA STARS	[DOT, 2014; Gelbart, 2003; Howell, 2014]
		Maritime Automated Identification system	[CTI, 2013; Lipton, 2006]
		California DMV	[Ingram, 1994]
		Healthcare.gov	[Alonso-Zaldivar, 2014; Kaeding, 2015; Morgan et al., 2013; Novet & Sullivan, 2014]
		Denver Airport Baggage system	[Coolman, 2014; de Neufville, 1994; Johnson, 2005; Myerson, 1994; Weiss, 2005]
		Boston Big Dig	[Gelinas, 2007; NTSB, 2007; Smith, 2010]
	Government acquisition	Seawolf Navy Submarine	[GAO, 1993; GAO 1994; Ward, 2014]
		Future Combat Systems	[Pernin, 2012; Schachtman, 2009]
		AMRAAM	[Mayer, 1993]
		Littoral Navy Ship	[Gallagher, 2014; Taubman, 2008]

[2] We place the unmanned space missions in Table 2.1 under project failures, since the primary effect of each event was failure to achieve mission objectives. None of these events involved loss of life or injury.

Table 2.1 continued

		DEA Plane	[OIG, 2016]
		F-35 Lightning II	[Chandrasekaran, 2013; Ciralsky, 2013; OIG, 2013; OIG, 2015]
		F-22 Raptor	[Vartabedian & Hennigan, 2013]
		V-22 Osprey	[Gertler, 2009; OIG, 1994; OIG, 2000; Thompson, 2007]
		Future Imagery Architecture	[Taubman, 2007]
	Space Mission	X-33 VentureStar	[Bergin, 2006]
		Mars Climate Orbiter	[Mishap Investigation Board, 1999; Oberg, 1999]
		NOAA N-Prime	[NASA, 2004]
		Solar and Heliospheric Observatory	[ESA & NASA, 1998]
		Mars Polar Lander	[JPL, 2000]
		Titan	[Pavlovich & Rea-Dix, 1999]
		Hubble	[NASA, 1990]
Accident	**Aerospace**	Challenger	[Rogers, 1986]
		Columbia	[Gehman et al., 2003]
		TWA 800	[NTSB, 1996a]
		Alaska 261	[NTSB, 2000]
		Colgan 3407	[NTSB, 2009]
		Aloha 243	[NTSB, 1989a]
		ValuJet 592	[Langewiesche, 1998; NTSB, 1996b]
		China 611	[ASC, 2002]
		Swissair 111	[TSBC, 1998]
	Energy	B.P. Texas City Refinery	[Baker et al., 2007; CSB, 2007a; Khan & Amyotte, 2007]
		Xcel Energy Hydro Plant	[CSB, 2007b]
		Kleen Energy	[CSB, 2010]
		Imperial Sugar	[CSB, 2008]
		Chernobyl	[IAEA, 1992; NRC, 1987; WNA, 2014]
		Fukushima Nuclear	[Action & Hibbs, 2012; Kurokawa, 2012]
		Three Mile Island	[Kemeny et al., 1979]
		Exxon Valdez	[NTSB, 1989b; Skinner & Reilly, 1989]
		Deepwater Horizon	[CSB, 2014; Graham & Reilly, 2011]

Table 2.1 continued

		Piper Alpha	[Cullen, 1990; Paté-Cornell, 1993]
		Westray Mine	[Cooke, 2003; Richard, 1997]
		Buncefield Oil Storage	[MacDonald, 2011; Newton, 2008]
		Pike River Mine	[Panckhurst, 2012]
		Upper Big Branch Mine	[McAteer, 2011; Stricklin, 2010]
		Cyanide Spill	[United Nations Environment Programme and the Office for the Co-ordination of Humanitarian Affairs, 2000]
		Bhopal	[Eckerman, 2005]
	Infrastructure	Walkerton	[O'Connor, 2002; Woo & Vicente, 2003]
		North Battleford	[Laing, 2002; Woo & Vicente, 2003]
		New Orleans Levee	[ASCE, 2007]
		Channel Tunnel	[CTSA, 1997]
		Lac-Mégantic Train	[TSBC, 2013]

We collected information on consumer products primarily from newspaper articles, and information on infrastructure projects, spacecraft mission failures, and government acquisitions from government reports (e.g. United States (U.S.) Government Accountability Office and NASA reports). We sourced information on accidents from investigation reports (e.g. U.S. National Transportation Safety Board (NTSB) and U.S. Chemical Safety Board (CSB) reports).

These sources vary in at least two ways. First, they differ in the level and extent of investigation. Newspaper articles, one of our primary sources for project failures, generally have the least depth and fewest details. Journalists are not necessarily trained to investigate accidents or failures, often only have limited time and resources to investigate, and the lengths of their articles may be dictated by editorial decisions. Additionally, organizations often choose how much information they provide to these journalists, as discussed in previous studies on project failures [Shore, 2008]. In contrast, many of the accident reports are lengthy and detailed. Accidents tend to be investigated by trained investigation specialists (e.g. aircraft accident investigators) or experts in technical fields (e.g. astronauts and academic researchers) with resources and access to sensitive information. GAO reports also appear to have no page constraints.

Second, different sources have different viewpoints or different purposes, which can affect the types of findings they make (for a discussion on pitfalls in accident investigations, see [Leveson, 2001]). Consider the Deepwater Horizon oil spill, which occurred just before the Deepwater Horizon drilling rig had completed drilling and plugging a well in the Gulf of Mexico. The crew on Deepwater Horizon was preparing the well for the oil platform when a sequence of events led to the plug failing, oil spewing from the well, and the drilling rig eventually burning down. Like all other United States chemical industry accidents, the accident was investigated by the Chemical Safety Board (CSB), a specialized independent agency that investigates industrial chemical accidents. Their 553-page report, which they published six years after the accident, comprises four volumes, detailing the (1) background on deep water drilling, (2) technical findings on failure of the Deepwater Horizon blowout preventer, (3) human and organizational factors contributing to the accident, and (4) regulatory oversight of offshore oil operations [CBS, 2014]. Then-president Barack Obama also commissioned an independent accident investigation committee comprised of politicians, engineers, and others to "determine what happened, why it happened, and explain it to Americans everywhere" [Graham & Reilly, 2011]. This report uses about 300 pages to discuss the history of the industry, causes of the accident, and the aftermath of the accident (the latter constitutes nearly a third of the report) and they published about eight months after the accident. The president-commissioned report was meant for a wider audience than the CSB report, which is apparent in the amount of discussion the authors dedicated to describing the context of the accident. For example, the president-commissioned report discusses the history of U.S. offshore drilling, including areas other than the Gulf of Mexico. As a result, the president-commissioned report contains less technical detail than the CSB report. For example, the committee pointed out that, "There is still much we do not know—for instance, the blowout preventer, the last line of defense against loss of well control, is still being analyzed" [Graham & Reilly, 2011, p. xi].

Newspaper articles are written to engage the reader and often intended to evoke emotional reactions. These articles may present accidents and failures in a more sensationalist way than official reports and omit details the journalist considers extraneous. For example, the president-commissioned report for the Deepwater Horizon accident noted that "BP's Macondo team had made numerous changes to the temporary abandonment procedures in the two weeks leading up to the April 20 [instructions]" [Graham & Reilly, 2011, p. 104]. When The New York Times reported the same finding, the authors said, "The [BP] executives were keen to keep the Horizon on track. In e-mails, BP managers—whose bonuses were heavily based on saving money and beating deadlines—kept asking when the well would be finished. […] BP has denied pressuring the Horizon's crew to cut corners, but its plans for completing the well kept changing, often in ways that saved time but increased risk" [Barstow et al., 2010]. The New York Times quote gives more context to the reader than the president-commissioned report quote, but the New York Times authors also sensationalized what happened in their interpretation.

2.2.2 Cause Extraction and Analysis

Our approach consists of five steps: (1) identifying findings in reports, (2) seeding our coding process with summary statements for findings from a subset of our cases, (3) applying the findings to a modified STAMP model to identify where in the design process they fall, (4) iteratively developing a coding scheme for the findings, and (5) coding the remaining findings to remove extraneous detail. We illustrate this process with the Deepwater Horizon oil spill and the F-35 Lightning II schedule and budget exceedances.

First, we extracted the findings of the Deepwater Horizon oil spill from the two accident reports [CSB, 2014; Graham & Reilly, 2011]. Table 2.2 shows a subset of the 33 findings for the Deepwater Horizon oil spill. We extracted the findings of the F-35 Lightning II budget and schedule exceedances from four newspaper articles and a U.S. Department of Defense report [Chandrasekaran, 2013; Ciralsky, 2013; OIG, 2013; OIG, 2015]. Table 2.3 shows a subset of the 46 findings for the F-35 Lightning II.

Table 2.2: Deepwater Horizon accident example statements and sources

#	**Report Extract (Finding)**	**Finding Summaries**
1	"The crew could not perform the negative-pressure test using the drill pipe; it would open the top of the drill pipe on the rig, bleed the drill pipe pressure to zero, and then watch for flow. […] the crew tried to bleed the pressure down to zero, but could not get it below 266 psi. […] [The site leader] then insisted on running a second negative-pressure test, this time monitoring pressure and flow on the kill line rather than the drill pipe. […] [The crew] made a key error and mistakenly concluded the second negative test procedure had confirmed the well's integrity." [Graham & Reilly, 2011, pp. 107-109]	Personnel inadequately addressed questionable test results.

Table 2.2 continued

2	"A blowout preventer can act as a barrier only if it is closed manually by the drilling crew or automatically as a result of a catastrophic event, such as a fire and explosion, which can trigger emergency backup systems. In manual operations, successful closure of the blowout preventer depends on several human decisions that must be made before a well kick can develop into a blowout. Otherwise, well pressures and well flow can exceed the design capabilities of the blowout preventer elements, leaving them unable to prevent or stop an active blowout." [CSB, 2014, p. 14]	Designers underestimated the severity of well blowout
3a	"The crew may have been distracted by other matters." [Graham & Reilly, 2011, p. 111]	Operations management gave operators too many tasks to perform at once.
3b	"As the crew conducted the test, the drill shack grew crowded. The night crew began arriving to relieve the day shift, and Harrell brought the VIPs through as part of their tour." [Graham & Reilly, 2011, p. 5]	Operations management gave operators too many tasks to perform at once.
4	"The laboratory personnel conducted several tests, including a foam stability test, starting an approximately April 13. The first test Halliburton conducted showed once again that the cement slurry would be unstable. The Commission does not believe that Halliburton ever reported this information to B.P." [Graham & Reilly, 2011, p. 101]	Laboratory personnel did not alert management about poor test results.
5a	"The [regulatory] agency's management shortcomings were underscored, and compounded, by lack of communication and inconsistencies among its three regional offices for the Gulf of Mexico, the Pacific, and Alaska. […] by acting in parallel fashion, with little coordination in decision-making and resource allocation, program implementation, regulatory interpretation, and enforcement policies became inconsistent, undermining the integrity of MMS's work." [Graham & Reilly, 2011, p. 78]	Regulatory body provided poor regulatory supervision of rig operations.
5b	The regulator "does not require industry to identify and manage all safety critical elements and tasks through defined performance standards, nor does it require assurance and verification activities to ensure a safety critical element is appropriate, available, and effective throughout its life cycle." [CSB, 2014, p. 16]	

Second, we modified the STAMP model [Leveson, 2004] to help us systematically identify where and when in the design process the finding occurred and used it as a framework for classifying the findings by organizational level. We chose the STAMP model over other accident models because it incorporates ideas from previous models (e.g., its hierarchical structure and system interactions), but introduces ideas important to understanding accidents and project failures (e.g. distinguishing between the system's development and operation, and having the phases of system development in the hierarchy). The STAMP model was initially used to describe ways in which systems could fail, including representing the systems' operating process as a control loop. We modified the model to help us depict a simple representation of a system, including its hierarchies and making a distinction between its development and operation. We removed the control loop process because our model only represents failures that occurred in the past, so we do not need to depict changing aspects of the system.

Figure 2.2 shows the model for the Deepwater Horizon case with the finding summaries 1 through 5 from Table 2.2 placed at appropriate locations on the model. For more information on how we modified the STAMP model and how we applied the findings to the model, see Sorenson [2015], and for access to the model applied to all of the failures we studied, see Aloisio & Marais [2017a].

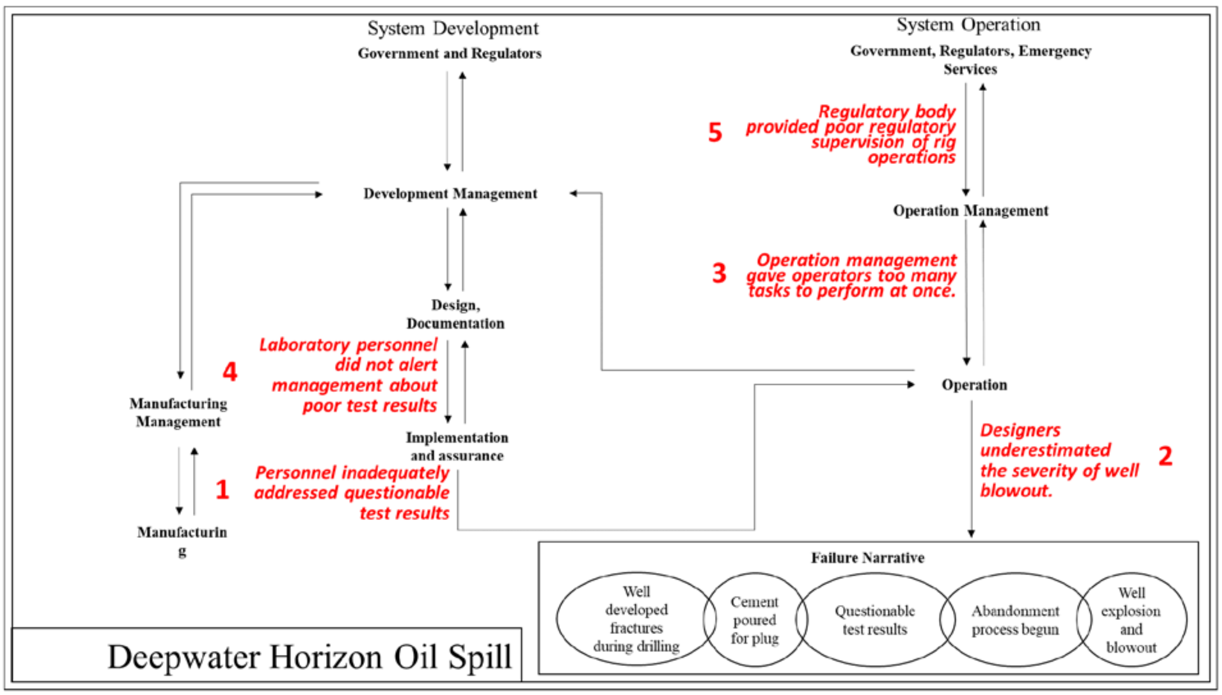


Figure 2.2: Deepwater Horizon causes applied to modified STAMP model

Next, we summarized each finding by discarding the specific details but retaining the defining information. For example, the first finding in Table 2.2 relates to problems with testing. The report refers to the “crew” and “site leader”; in our finding summary, we use the general term “personnel”. The report also provides the specific details on what the crew did incorrectly—in this case they conducted a test that gave unfavorable results and decided to redo the test in a way that made favorable results more likely, rather than determining why the first test gave unfavorable results. We summarized this finding as “insufficiently addressed questionable test results”. Many reports refer to the same instance of a particular problem more than once—for example in a body chapter and also in the conclusion. Cases with more than one report (e.g., Deepwater Horizon) also resulted in more than one extract referring to the same instance of a particular problem, as indicated for example in rows 5a and 5b of Table 2.2. Reports may also refer to different instances of the same problem, as indicated for example in rows 3a and 3b of Table 2.2. In Table 2.2, rows 5a and 5b discuss two different regulator shortcomings. We therefore counted these excerpts as two findings. In contrast, rows 3a and 3b both refer to the same instance of the same problem—accordingly we counted these excerpts as one finding.

The reports vary in how they specify the parties involved in a particular finding. Some reports contain extensive details, including names and roles (e.g. the Walkerton water contamination accident discusses the actions of a particular manager [O'Connor, 2002]). Some reports specify the roles of people involved, such as the Deepwater Horizon report referring to the "crew" and "site leader" (see Table 2.2). Some reports do not specify names or roles, but provide other information that allowed us to infer the roles in all such cases. For example, consider the third finding in Table 2.2, in which the oil rig crew was distracted by a VIP tour while conducting an important test in a small control room. We were able to infer from the report that the persons responsible for bringing the VIPs were in an operations management role (the person giving the tour was described as "the top Transocean man on the rig" [Graham & Reilly, 2011, p. 5]. Similarly, consider the first cause in Table 2.3, in which Lockheed Martin poorly supervised its "suppliers" and "subcontractors". Since these terms tend to be used ambiguously in failure reporting, we created one umbrella term, "supplier". This term describes an organization independent from development management (e.g. Lockheed Martin) that provides goods (e.g. a pump) or services (e.g. design work). We followed a similar process to summarize the remaining 25 Deepwater Horizon accident findings and 25 F-35 project failure findings.

Table 2.3: F-35 project failure example statements and sources

#	**Report Extract (Finding)**	**Finding Summary**
1	"Our assessment identified that Lockheed Martin neither adequately provided review or [sic] approved of engineering change submittals made by Lockheed Martin's critical suppliers." [OIG, 2013, p. 12]	Development management poorly supervised suppliers.
2a	"Technological innovation, including heavy reliance on computer simulation, which could take the place of real-world testing, would keep costs down. […] Building an airplane while it is still being designed and tested is referred to as concurrency. In effect, concurrency creates an expensive and frustrating non-decision loop: build a plane, fly a plane, find a flaw, design a fix, retrofit the plane, rinse, repeat." [Ciralsky, 2013]	Computer simulations were inadequate tests to identify design problems.

Table 2.3 continued

2b	"Pentagon officials accepted Lockheed's claim that computer simulations would be able to identify design problems, minimizing the need to make changes once the plane actually took to the sky. […] [But] early tests uncovered flaws unnoticed by the computer simulations." [Chandrasekaran, 2013]	Computer simulations were inadequate tests to identify design problems.
3a	"The Air Force, Marines and Navy all sought additional modifications to meet their needs, reducing commonality among the three models. A bigger problem was the fundamental concept of building one plane, with stealth technology, that could fly as far and fast as the Air Force wanted while also being able to land on the Navy's carriers and take off vertically from Marine amphibious assault ships." [Chandrasekaran, 2013]	Development management tried to please too many customers in one limited design.
3b	"From the outset, critics have worried that by trying to meet so many missions for so many masters, the Joint Strike Fighter would end up being […] a 'jack of all trades, and master of none.' Take the matter of stealth technology, which helps an airplane elude detection. […] it doesn't serve much purpose in a Marine Corps environment." [Ciralsky, 2013]	Development management tried to please too many customers in one limited design.
4	"We identified that Lockheed Martin did not maintain mission systems requirements traceability to the software-level requirements. […] Untraceable requirements cannot be verified for impact on system performance." [OIG, 2013, p. 13]	Development management did not keep track of requirements.

Some investigation bodies record accidents using a coding system, such as the NTSB's method for investigating aviation accidents [NTSB, 1998]. This type of system allows the investigators to have a baseline from which to analyze multiple accidents at once. The NTSB coding system facilitates analysis of overall trends in accident causation. Here, we coded each statement into an "actor-causal action-object" structure, where the actor is the person (or group of people), the causal action is what they did, and the object provides detail about the causal action. Figure 2.3 shows an example for two findings, from the Deepwater Horizon accident and the F-35 project failure. In both cases, testing was inadequate in some way, so we created a "subjected equipment to inadequate testing" causal action. In the Deepwater Horizon case, it was the personnel conducting the test who did not adequately investigate the questionable test results. Had they done so, they would likely have realized that they needed to redo the test. In contrast, on the F-35, development managers requested a form of testing (computer simulation) that was insufficient. Thus, we assigned responsibility to the development managers, rather than to the engineers conducting the simulations. The objects for each statement, "safety testing" and "development testing", identify the specific type of testing.

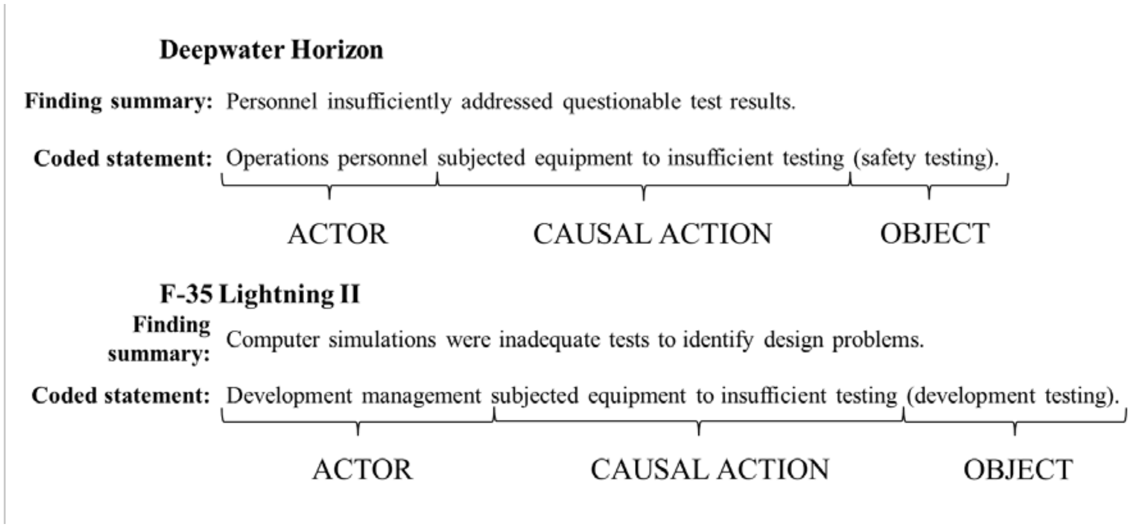


Figure 2.3: Actor-causal action-object structure for findings in different failures

When a particular finding involved more than one actor, causal action, or object, we assigned additional unique actor-causal action-object codes to the finding to illustrate all facets of the finding. Figure 2.4 shows an example of a finding from the Westray Mine collapse to which we assigned two coded statements.

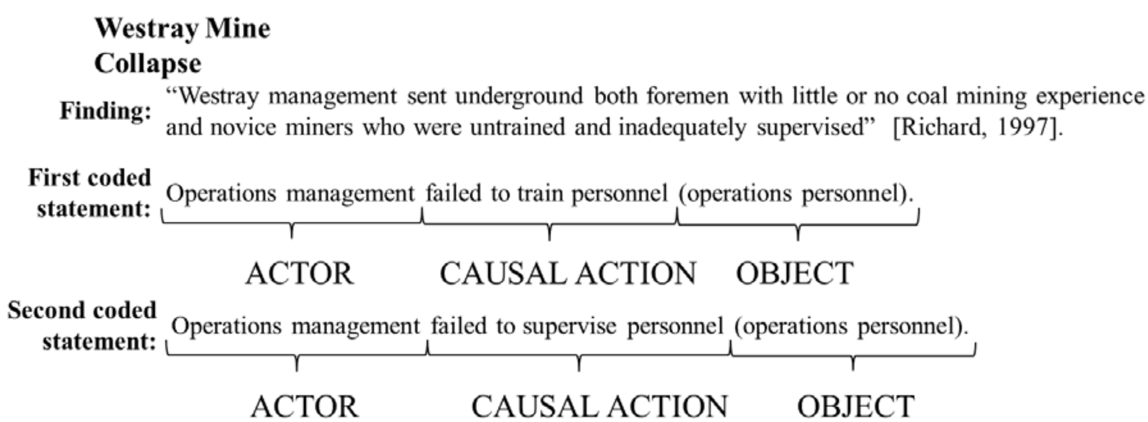


Figure 2.4: Actor-causal action-object structure for findings with multiple coded statements

We identified a total of 966 findings, which we represent using a set of 23 causal actions, 9 actors, and 119 objects. For our complete dataset, see Aloisio & Marais [2017b]. Each causal action is associated with at least one object; for instance, "subjected to inadequate testing" has objects describing five types of testing: acceptance, development, quality, reliability, and safety testing. Other causal actions have more abstract objects. For example, “used inadequate justification”, has objects like “acquisition” and “hiring”.

Each actor is associated with a certain role, and inevitably leads to some actors being more tied to certain causes than others. For example, regulators almost exclusively are involved in enforcing inadequate regulations. Table 2.4 contains the 9 actors and their descriptions.

Table 2.4: Actor descriptions and examples

Actor	**Description**
Designers	This group is responsible for making specific design decisions (such as design engineers).
Development Management	This group is in contrast to designers in that they make more overarching decisions, for instance about the direction or purpose of the project. An important aspect of this group is that they manage others.
Government (Customer)	Governments, such as the United States Government, commission defense projects and have a special role in that they identify requirements for a project but do not necessarily dictate the specifics of the project.
Government (Regulator)	The regulator is typically an independent agency tasked with enforcing rules and regulations associated with project design, implementation and operation (such as the Federal Aviation Administration (FAA) in the United States, which enforces regulations pertaining to aircraft design and operation).
Government (Lawmaker)	The lawmakers make laws or allocate government funds regarding government-run institutions. In our dataset, this actor was involved in allocating funds to a government-run water treatment plant.
Manufacturing Management	This group oversees manufacturing on a project.
Operations Management	This group oversees project operations.
Operations Personnel	This group is any person in an organization whose role is neither defined by managing others nor designing the system.
Supplier(s)	The supplier has a unique role in project development in that they are an independent organization that provide a service. Their culture and values could be completely distinct from the organization commissioning their services.

2.3 Cause Discussion

In this discussion, we focus on the causal actions. While it is possible that failures could be predominantly caused by a particular person or group of people (e.g., errors of various types by the maintenance division), our data does not support such an assumption. Instead, we found that different actors tended to make similar mistakes (e.g., both maintenance technicians and design engineers kept poor records). Additionally, these actors made similar mistakes on different "objects" (e.g., designers kept poor records of the design process as well as of the test process). The "causal action" from each statement gives insight into what happened, rather than focusing on which particular person or group was responsible or to what specifically the causal action pertained (e.g. the type of testing). In the remainder of this paper, we will simply refer to "causal actions" as "causes".

Accidents and other project failures share many causes. Which causes are most reported in accidents and project failures? Are some causes reported more in accidents than in project failures, and vice versa? To answer these questions, we define the presence of each $cause_i$ as the proportion of accidents (or project failures) that contain at least one mention of that cause:

$$presence(cause_i) = \frac{\sum_{k=1}^{k=N} TRUE(cause_i | failure_k)}{N} \quad (1)$$

Where $failure_k$ is the k^{th} accident or project failure and N is the number of accidents or project failures. For example, we identified failed to train at least once in 19 of the 30 accidents, thus its presence in accidents is 63%. This cause occurred at least once in 4 of the 33 project failures, thus its presence in project failures is 12%.

2.3.1 Dissimilar Causes

In this section, we discuss each dissimilar cause. We present a definition for each cause, examples from accidents and project failures, and posit reasons for the differences in presence for the causes. Figure 2.5 shows the causes that have dissimilar presences for accidents and project failures.

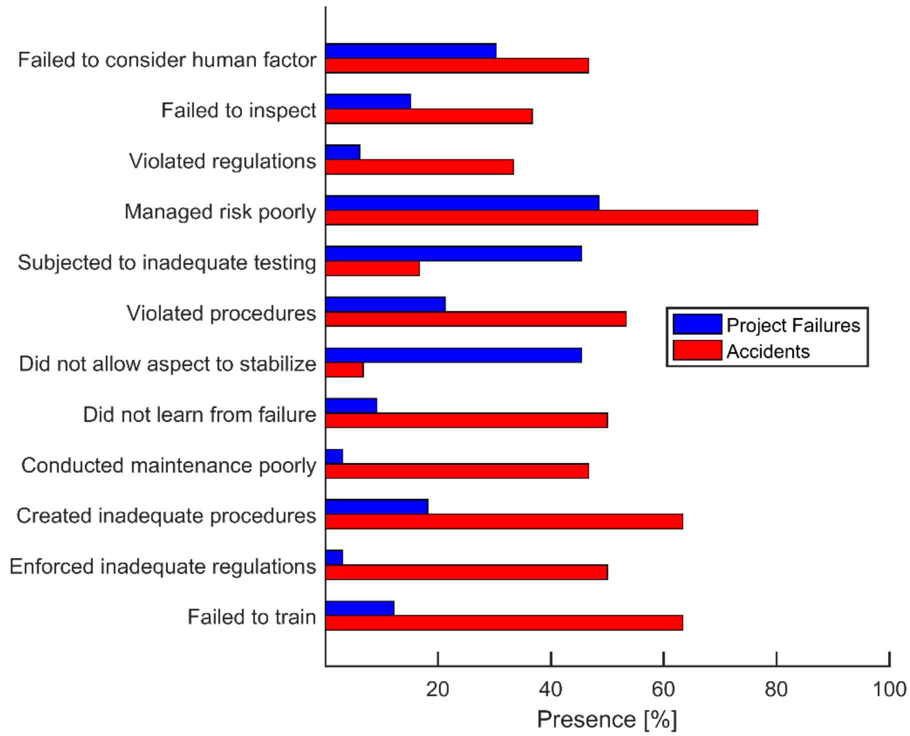


Figure 2.5: Causes with dissimilar presences with increasing difference

Causes reported more often in project failures than in accidents

Two causes are reported more often in our project failures cases than in the accidents: subjected to inadequate testing, and did not allow aspect to stabilize.

Subjected to inadequate testing: One or more actors in the organization subjected a component or subsystem to inadequate testing. This causal action captures inadequate tests as well as adequate tests performed inadequately.

This cause is present in only 5 of the 30 accidents we studied, possibly because many of the accidents involved older systems that were well past their development stages. Accident investigators may have had trouble finding development documentation about how the system was tested, making it difficult to identify testing problems. Additionally, some systems, such as the Three Mile Island nuclear power plant, were made by one organization and then operated by a different organization [Kemeny et al., 1979], which could make it difficult for accident investigators to find information on testing. We also found documentation on testing problems in the two Shuttle accidents and two of the aircraft accidents. Both the aviation and space industries are subject to strict rules and oversight. Thus, problems during development are likely more easily accessible to investigators. We found this cause in 45%, or 15 of our 33 project failures. For example, it occurred in 9 of the 16 government acquisitions and mission failures, for which transparency to taxpayers is also a concern.

Did not allow aspect to stabilize: Actor(s) in the organization did not allow a system aspect like personnel, design, or requirements to stabilize before moving forward with the project.

This cause is present in only a few accidents (7%) but half of project failures. It is common in projects that proceeded with a design before dealing with problems like quality issues. For example, construction on the Navy Seawolf submarine began before the design was complete, which led to a lot of redesign, and eventually construction rework [Kennedy & Conyers, 1994]. It may be easier to motivate resolving safety issues before proceeding, because lives may be involved, and because there are more regulations and procedures. Waiting to resolve non-safety-critical quality concerns may be harder to motivate in a competitive market, especially when concerns are ambiguous. Many of the consumer electronics project failures we studied were released before software bugs were resolved. An example of problems with “personnel instability” is in the Texas City refinery explosion, in which “process safety leadership appeared to have suffered as a result of high turnover of refinery plant managers” [Baker et al., 2007, p. 59].

Causes reported more in accidents than in project failures because accident investigations are more detailed and thorough

Nine causes were reported more often in our accident cases than in the project failures. Four of these causes tended to be identified more often in the more thorough accident investigations: failed to train, created inadequate procedures, violated procedures, and failed to inspect. We suspect that their higher presence is an artifact of accident investigations generally being more detailed and thorough than project failure investigations, as discussed in Section 2.2.1.

Failed to train: Actor(s) in the organization failed to train other actors in the organization, such as operations personnel or maintenance personnel.

This cause is reported in a majority (63%) of accidents but in relatively few (12%) project failures. In both accidents and project failures, the training deficiencies were mostly for operations personnel. For example, in the Westray Mine collapse, the "miners, supervisors, and underground tradesmen at Westray were not provided with adequate training in safe underground work practices" [Richard, 1997]. We suspect that the infrequency of this cause in project failures is due to differences in project failure and accident investigations. As discussed earlier, journalists may be given less access to organizations' failures and thus be less likely to identify detail items like training records. This hypothesis is supported by the fact that most of the training deficiencies we identified in project failures occurred in mission failures. These failures are investigated and reported by space agencies like NASA, which has access to its own training records.

Created inadequate procedures: Actor(s) in the organization developed a deficient procedure, for instance maintenance, manufacturing, or emergency procedures.

This cause is also reported in a majority (63%) of accidents but in few (18%) project failures. Accident investigators identified many cases of inadequate emergency and safety procedures, while project failures are less likely to have procedures related to emergency and safety. For instance, in the Alaska Airlines flight 261 crash, when the horizontal stabilizer did not respond properly, the pilot attempted different control configurations until the faulty jackscrew completely gave way and the aircraft nose-dived into the ocean. The NTSB criticized the emergency procedures, stating: “Without clearer guidance to flight crews regarding which actions are appropriate and which are inappropriate in the event of an inoperative or malfunctioning flight control system, pilots may experiment with improvised troubleshooting measures that could inadvertently worsen the condition of a controllable airplane” [NTSB, 2000, p. 140].

Violated procedures: Actor(s) in the organization violated a procedure pertaining to the system, such as a maintenance or operation procedure.

This cause is reported more than twice as often in our accidents (53%) as it is in the project failures (21%). While both accidents and project failures involved violation of operating procedures, some of the government acquisition projects violated acquisitions procedures. For example, in the Healthcare.gov website construction, the Centers for Medicare & Medicaid Services (CMS) “did not consider previous contractor performance for many bids even though federal contracting rules require it” [Kaeding, 2015]. Accident investigators also frequently criticized maintenance procedure violations.

Failed to inspect: Actor(s) in the organization failed to inspect a crucial component.

This cause was reported in 40% of our accidents and only 15% of the project failures. It was reported in a majority of the aircraft accidents. For example, in the Aloha Airlines flight 243 accident, aircraft inspections did not identify fatigue damage. In this case, the NTSB pointed out that: “Of additional concern was Aloha Airlines’ practice of inspecting the airplane in small increments. Limited areas of the airplane were inspected during each work package and this practice precluded a comprehensive assessment of the overall structural condition of the airplane” [NTSB, 1989a, p. 52].

Causes reported more in accidents than in project failures because safety is more regulated

Causes related to regulations are more likely to be found in accidents because safety is more regulated, than, for example, cost estimation and management. Thus, we were not surprised to find violated regulations and enforced deficient regulations in more of our accidents than project failures.

Violated regulations: Actor(s) in the organization violated a regulation pertaining to the system. This cause was reported in a third (33%) of the accidents we studied, across all types of systems. For instance, in the Exxon Valdez oil spill, the “master’s decision to leave the third mate in charge of the navigation watch was contrary to Federal regulations and Exxon policy and was improper given the course of the vessel, the uncertain extent of the ice conditions, the proximity of a dangerous reef and the fact that the third mate did not have the required pilotage endorsement” [NTSB, 1989b]. The two instances of this cause in project failures are in the DEA plane acquisition (in which the DEA did not comply with regulations while purchasing the aircraft) and the Future Imagery Architecture satellite (in which a supplier made some parts with tin, which is unsuitable for use in space) [OIG, 2016; Taubman, 2007].

Enforced inadequate regulations: A regulator (e.g., the FAA) enforced deficient regulations. This causal action captures writing deficient regulations as well as implementing regulations poorly.

This cause was reported in half of accidents (50%) across all types of systems, as for example in the Swissair 111 crash, where “less stringent material flammability standards were applied to those materials that were intended for use within the pressure vessel but that were outside the occupied areas” [TSBC, 1998, p. 107]. The one instance of this cause in project failures is in the Ford Explorer quality issues, where the investigation noted that: “Essential safety standards are severely out of date, were scrapped or delayed in the Reagan years, or are prohibited by law because of industry lobbying” [Claybrook, 2000].

Accidents are investigated more thoroughly and their investigators are more focused on risk
It is likely that the higher presence in accidents of the causes managed risk poorly, did not learn from failure, and failed to consider human factor is due to better, more thorough investigations and to a greater focus on the role of risk.

Managed risk poorly: Actor(s) in the organization failed to identify, assess, formulate, or implement a proper mitigation measure.

This cause is present in 77% of accidents, but in only 42% of project failures, even though it is likely that many project failures involve poor risk management. For example, writing about the Boeing 787 Dreamliner development, Denning [2013] notes that the "cultural and language differences and the physical distances involved in a lengthy supply chain create additional risks. Mitigating them requires substantial and continuing communications with the suppliers and on-site involvement, thereby generating additional cost. Boeing didn't plan for such communications or involvement, and so incurred additional risk that materialized". The project failure presence is likely an underestimate. While many of the project failure risk management criticisms focused on identifying and assessing risk, the accident risk management criticisms focused on mitigating risk. This difference may indicate that people involved in project failures "did not know what they did not know" and people involved in accidents knew about the risks but did not take the proper steps to mitigate them.

Did not learn from failure: Actor(s) in the organization did not take past failures into account and a similar problem occurred.

This cause is reported in 50% of our accidents but in only 3 (9%) of our project failures. This difference may be indicative of the difference in how organizations cooperate to prevent different types of failures. In general, organizations are more willing to share findings on safety, including lessons learned. For example, the Aviation Safety Reporting System (ASRS) is a system that collects voluntarily submitted aviation safety incidents from people in the aviation community, like pilots and controllers [NASA, 2015]. In contrast, organizations are much less likely to share lessons learned on preventing budget or schedule overruns.

Failed to consider human factor: Actor(s) in the organization failed to consider a human factor in system development. This causal action describes, for example, failing to consider human factors in specifying procedures or physical design.

We found this cause in 47% of our accidents and 27% of project failures. For both project failures and accidents, policies leading to worker fatigue was a problem. For instance, in the Texas City refinery explosion, "[s]ome employees had worked up to 30 days of consecutive 12-hour shifts. The reward system within the site encouraged this extended working period without consideration of fatigue. There were no clear limitations on the maximum allowable work periods without time off" [Baker et al., 2007, p. 87]. Equipment design was a problem in many aircraft accidents. For example, in the Swissair flight 111 crash: "[i]n the deteriorating cockpit environment, the positioning and small size of these [standby] instruments would have made it difficult for the pilots to transition to their use, and to continue to maintain the proper spatial orientation of the aircraft" [TSBC, 1998, p. 254].

Causes reported more in accidents than in project failures because of the nature of the project failures we studied

Finally, many of the project failures we studied occurred before the systems had matured through their design cycles. Thus accidents had more instances of the cause *conducted maintenance poorly*.

Conducted maintenance poorly: Actor(s) in the organization failed to perform maintenance on a component or subsystem.

This cause was reported in 47% of our accidents and only one of the project failures, occurring less often in the aerospace accidents and more often in the energy and infrastructure industry accidents. For instance, in the Three Mile Island nuclear accident, "[r]eview of equipment history for the 6 months prior to the accident showed that a number of equipment items that figured in the accident had had a poor maintenance history without adequate corrective action" [Kemeny et al., 1979, p. 47]. The single instance of this cause in project failures is in the Hubble spacecraft mirror flaw, in which the equipment used to manufacture the mirror (and responsible for the flaw) had been poorly maintained [NASA, 1990].

2.3.2 Similar Causes

In this section, we discuss each similar cause. We present a definition for each cause, examples from accidents and project failures, and posit reasons for the similarities in presence for the causes. Figure 2.6 shows the eleven similar causes. These eleven causes can all apply to any stages in projects' lifecycles. On average, the project failure presence for similar causes (42%) was about twice that for dissimilar causes (21%). Similarity seems to be at least in part a side-effect of causes being identified more often, perhaps because they are easier to identify. All eleven of these similar causes do not require an understanding of risk, or of the role of regulations, procedures, and training in failures.

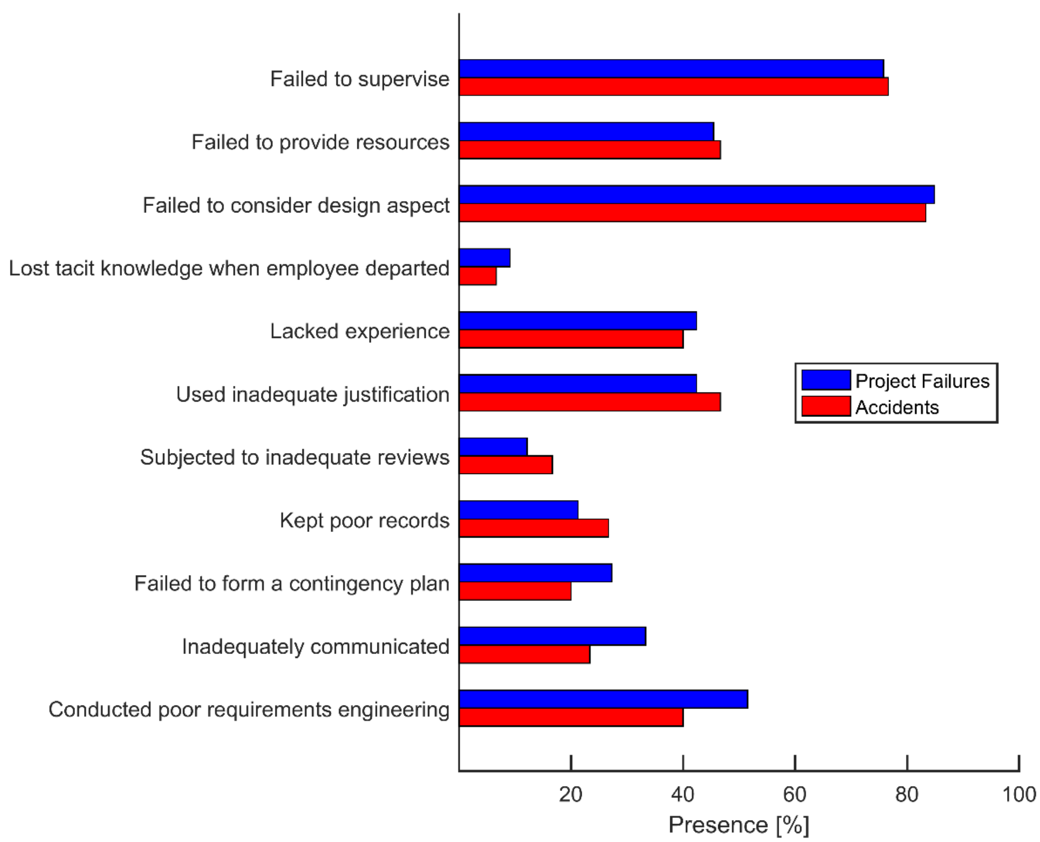


Figure 2.6: Causes with similar presences with increasing difference

Management errors

Management errors like *failed to supervise* and *failed to provide resources* were reported equally often for both project failures and accidents.

Failed to supervise: Actor(s) in the organization failed to supervise people or a process properly. This cause was reported equally often in project failures (76%) and accidents (77%) across all types of industries. For example, the GAO investigation of the Healthcare.gov website quality issues found that: "the administration lacked 'effective planning or oversight practices'" [Alonso-Saldivar, 2014].

Failed to provide resources: Actor(s) in the organization failed to provide adequate resources to a department; for instance, maintenance, marketing, or safety.

This cause was reported in 48% of our project failures and 47% of accidents. For example, during the Xbox 360 development; "[e]ven though Microsoft's leaders knew their quality wasn't top notch, they did not ensure that resources were in place to handle returns and quickly debug bad consoles" [Takahashi, 2008].

Keeping track of information

Causes that required keeping track of information, like *lost tacit knowledge when employee departed* or *kept poor records* were reported similarly in accidents in project failures.

Lost tacit knowledge when employee departed: Personnel quit, were moved to a different project, or retired, and the organization failed to sustain the knowledge base without these persons. This cause occurred in only two project failures and two accidents. On the Xbox 360 development, experienced engineers were spread too thinly over too many projects [Takahashi, 2008]. The Future Imagery Architecture spacecraft lost senior staff members, perhaps through retirement [Taubman, 2007]. At the Westray mine, the most experienced miners quit because they recognized the terrible conditions; less experienced ones stayed because they needed the money. And in North Battleford, the experienced plant foreman at the water treatment plant retired and was replaced by an inexperienced one. It is somewhat surprising that we found this cause so infrequently, given the attention the concept of tacit knowledge receives in management literature.

Kept poor records: Actor(s) in the organization kept poor records of a process, such as maintenance.

This cause was reported in 21% of our project failures and 27% of accidents. This cause often appeared in government acquisitions and mission failures, for which there are systems and organizations dedicated to ensuring accountability to taxpayers, like the US Government Accountability Office (GAO) and the US Office of the Inspector General (OIG). For example, in its review of the F-35, the OIG "identified that Lockheed Martin did not maintain mission systems requirements traceability to the software-level requirements. […] Untraceable requirements cannot be verified for impact on system performance" [OIG, 2013, p. 13].

This cause was reported slightly more often in aircraft accidents than the other types of accidents we studied, probably because the FAA requires the commercial aviation industry to keep detailed records, and it is obvious when these records are missing. For instance, in the TWA flight 800 crash, the "Safety Board found evidence of repairs accomplished near fuel quantity indicator system wire routing areas for which no associated maintenance records were found" [NTSB, 1996, p. 52].

Design criticisms

Design criticisms like *failed to consider design aspect* and *conducted poor requirements engineering* were equally pervasive in accidents and project failures.

Failed to consider design aspect: Actor(s) in the organization failed to consider an aspect in the system design. In many cases, this causal action describes a design flaw, such as a single-point failure or component compatibility.

This cause was reported the most often in both our project failures (85%) and accidents (85%), in part because it covers a wide range of design criticisms (which are captured by the objects, see

Figure 2). Some designers did not consider component compatibility when designing their system, like in the Texas City refinery explosion where the "size of the blowdown drum was insufficient to contain the liquid sent to it by the pressure relief valves. The blowdown drum overfilled and the stack vented flammable liquid to the atmosphere, which fell to the ground and formed a vapor cloud that ignited" [CSB, 2007, p. 24]. Other designers did not consider single-point failures in their system, which is a frequent criticism for safety in systems but also in the case of the F-35: "Commonality simply meant that the three F-35 variants would share portions of high-cost components like the airframe, the avionics, and the engines. This was supposed to help ensure that the plane was 'affordable.' […] Commonality, even at this reduced level, has unintended consequences. When a crack in a low-pressure turbine blade was discovered in an air force F-35A engine earlier this year, Pentagon officials took the only responsible course, given that the part is used in all models: they grounded the entire fleet of F-35s, not just the ones flown by the air force" [Ciralsky, 2013]. Other design criticisms include failing to consider customer needs, system interactions, and changing environments within the system.

Conducted poor requirements engineering: Actor(s) in the organization did not lay out the needs, attributes, capabilities, characteristics, or qualities of the system well.

This cause was reported a bit more often in project failures (52%) than in accidents (40%). Most of the criticisms for project failures were that they did not plan the performance requirements well, as for example in the V-22 Osprey, where the "V-22's less than 400-hour engine service life fell short of the 500-600 hours estimated by program management. The program office noted that the contract does not require a specific service life to be met" [Gertler, 2009, p. 9]. Many accidents involved poor project planning and safety requirements, as Paté-Cornell stated in her analysis of the Piper Alpha oilrig fire: "Some of these additions [to the rig] apparently interfered with the proper functioning of safety features: external reinforcements on module C, for example, prevented adequate functioning of the blast relief. […] The result was that safety features that may have been adequate in the beginning became inadequate for this new layout, with new couplings and higher risks of accident that may not have been realized (or sufficiently questioned) at the time when the additions were made" [Paté-Cornell, 1993].

People problems like *inadequately communicated* and *lacked experience* were also reported with similar presence in accidents and project failures.

Inadequately communicated: Actor(s) in the organization failed to communicate with each other such that personnel were confused with the information they were given, had to "fill in the gaps" in the information they were given, or not notified about important information at all.

This cause was reported in 33% of our project failures and 23% of accidents. For example, the Navy's March 1992 assessment of the Seawolf Navy Submarine schedule and budget overruns showed "an apparent incomplete coordination with industry and inadequate notification to and consultation with industry regarding major changes in Seawolf specifications as required by the Naval Sea System Command's specification process" [Kennedy & Conyers, 1994, p. 6].

Lacked experience: Actor(s)' lack of experience or knowledge led to the failure. For example, an inexperienced manager who was placed in charge of a large project.

This cause was reported in 45% of our project failures and 40% of accidents. In project failures overall, investigators often blamed a lack of management experience; when we narrow the set to mission failures, investigators often blamed a lack of technical experience. For example, the Solar and Heliospheric Observatory (SOHO) spacecraft lost communication with NASA because one of the people who modified the software for one of the spacecraft's gyroscopes lacked technical knowledge of the system, and thus did not include a logical step in an emergency mode for the software [ESA & NASA, 1998]. In accidents, this cause was reported across all industries, with slightly more emphasis on a lack of technical experience rather than a lack of management experience.

Insufficient information

Accidents and project failures both moved forward with operations without sufficient information, which we described with the causes *used inadequate justification*, *subjected to inadequate reviews*, and *failed to form a contingency plan*.

Used inadequate justification: Actor(s) in the organization used inadequate justification for a decision.

This cause was reported in 48% of our project failures (48%) and 43% of accidents. In project failures, this cause was a criticism for government acquisitions and consumer products, like in the Merck Vioxx medication recall: "Our research […] found that compared to naproxen, a commonly used over-the-counter anti-inflammatory drug with similar benefits, Vioxx has a five times greater heart attack risk. In response, Merck claimed that early conclusions about the risk were flawed, and attributed the comparatively high heart attack rates to an unproven protective effect of naproxen" [Topol, 2004]. For accidents, this cause was a criticism for all industry types, but most notably for aerospace accidents.

Subjected to inadequate reviews: Actor(s) in the organization did not review documentation or other work sufficiently to capture errors and deficiencies.

This cause was reported in few project failures (12%) and accidents (17%). For project failures, this cause was reported exclusively for government acquisitions and mission failures, possibly because their systems usually have formal review structures in place. For example, in the Mars Polar Lander loss, the JPL Special Review Board [2000] noted that, "[i]n the case of the Propulsion Subsystem, the thermal control design interfaces were not mature enough to evaluate at the [Critical Design Review] CDR. A delta review should have been held but was not. Such a review could have discovered the problems experienced in flight". This cause was reported across all types of accidents.

Failed to form a contingency plan: Actor(s) in the organization failed to form a contingency plan to implement if an unplanned event occurred.

This cause was reported in 24% of our project failures and 20% of accidents, across all industry types except for aircraft accidents. This absence is most likely because pilots are trained in extensive contingency plans for a wide range of possible emergencies. An example of this cause is in the Exxon Valdez oil spill: “Government and industry plans […] did not assume a spill of the magnitude of the Exxon Valdez spill and the Alyeska Plan did not provide sufficient detail to guide the response” [Skinner & Reilly, 1989, p. 8].

2.3.3 Comparison to Other Failure Studies

Our analysis is novel in that we use accident research, a relatively untapped resource in project failure research, to understand project failures better. Thus, an important question to answer is: Did comparing accident and project failure causes provide new insight into project failure causation? This section briefly compares our findings with other studies on project failure and accident causation.

Studies that looked at statistical data on failures

Some failure studies gathered data on a large number of failures to see which causes are the most frequent. Newman [2001] analyzed fifty space system failure case studies from U.S., Russian (and Soviet), and French space agencies and show in what general area each of these failures had problems (for example, design flaws and inadequacies in training and experience). The author then discusses the areas that had the most clustered data and how these problems could be improved in a general manner. We cited similar “problem areas” in that our study criticized poor training and design flaws.

Konstandinidou et al. [2011] did a much larger study and took a statistics-based analysis approach. The authors gathered data on 1,112 reported incidents in the Greek petrochemical industry over a seven-year period and even included “near misses” in their study, which are similar to accidents in that they are unplanned events but do not result in loss like an accident. These authors grouped problems into categories, such as “inadequate procedures”, “inadequate training”, and “lack of communication” and showed how many incidents and accidents suffered from each of these problems. This study is very similar to ours in that they also looked at the presence of different problems in the accidents and incidents they studied, but since they studied such a comprehensive number of events their study has significant statistical power. However, the aim of our research is not to show how each of the causes we identified led to the severity of the failures we studied using a statistical analysis, but rather to prove that these causes may combine in different ways to manifest in different failures, such as both accidents and project failures.

Project management studies

Other studies have taken approaches to studying failures based in project management. Pinto & Mantel [1990] sent questionnaires to members of the Project Management Institute (PMI) that asked the respondents to define a project's success or failure based on 13 items, such as "communication" and "top management support". These authors used a statistical analysis on the responses they received to determine the industry's consensus on what exactly can lead to a project failing or succeeding. While our study does have input from systems engineers, it does not have the statistical power and thus an "industry consensus" on what can lead to project failures.

Williams et al. [2012] synthesized project management literature to show how different project assessments, such as project reviews and audits, could identify "early warning" signs in projects, which are signals of a project experiencing a negative event in the future. The authors then interviewed project management experts on how they do or do not implement these project assessments, and followed up their study by analyzing eight case projects to see whether they had used project assessments to successfully identify early warning signs. This study focused heavily on what could go wrong from a project management perspective, such as poor project definition, poor communication, and a lack of documentation, which our study also identified.

Shore [2008] looked at project management using a behavioral view and proposed that understanding systematic biases (in human decision-making) could help diagnose project failure. The author defined nine systematic biases, such as "groupthink" and "sunk cost" that lead to project failures. Our results are quite different from Shore's [2008] findings, despite both of our studies discussing many of the same failures. Our findings are complementary. Shore focused on identifying systematic biases within organizations' cultures, while we attempted to extract all the identified causes of each failure, and built our coding system based on the terminology used by the investigators. Our study purposely did not attempt to identify or infer causes beyond those cited in the reports.

2.4 Linking Causes and Recommendations

Project failure reports rarely contain recommendations. Only one of the project failures we studied contained recommendations (the Drug Enforcement Administration (D.E.A.) plane [OIG, 2016]), and these recommendations do not address the underlying problems that led to the failed acquisition. In contrast, most large accident investigations include extensive recommendations on how to prevent future accidents. Since we have found that accidents and project failures share many causes, recommendations from accident investigations are potentially also applicable to project failure prevention.

Figure 2.7 describes our approach to coding and analyzing the recommendations from accident reports, using excerpts from the Imperial Sugar Refinery Accident report [CSB, 2008]. First, we linked the accident report findings to the corresponding recommendations. Some accident reports explicitly link recommendations to specific findings (e.g., the Space Shuttle Columbia accident report [Gehman et al., 2003]), but most of the reports do not. For example, NTSB reports have a section labeled “findings” followed by a section labeled “recommendations”, but in general there is no explicit link to the recommendations from the findings. One of the reports did not make any recommendations at all (the Bhopal accident [Eckerman, 2005]) and others made only a few recommendations, often addressing only a subset of the findings.

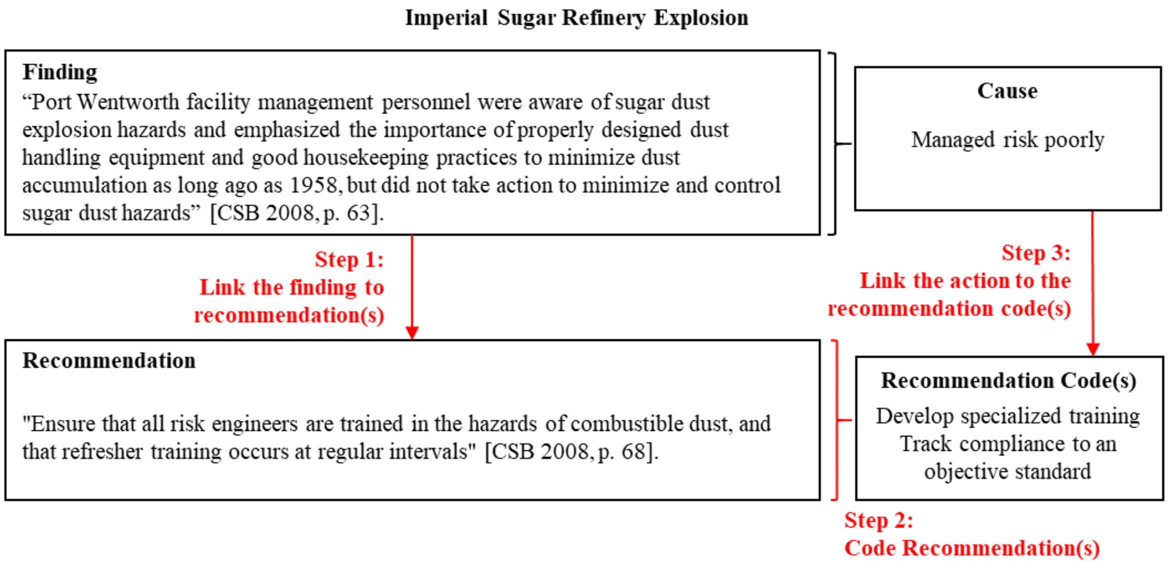


Figure 2.7: Recommendation coding and linking diagram

We used a similar approach to the cause coding to code the recommendations. Some findings had multiple recommendations that spanned many ideas, so a single cause could have more than one recommendation, and hence potentially more than one recommendation code. In Figure 2.5, we connected the finding to a single recommendation, which we described using two recommendation codes because it contains two distinct ideas. In total, we identified 16 recommendation codes, as shown in Table 2.5.

Table 2.5: Recommendation code definitions

Recommendation	**Definition**
Conduct random and independent evaluations	Perform an evaluation like an inspection or audit on a component, organization, or person, and do it randomly, often, and by an independent organization or party.
Develop a comprehensive and rigorous test	Develop a test that includes all possible regimes, equipment, and situations, and is stricter than what is minimally necessary (e.g. to a certain factor of safety).
Develop specialized training	Develop training to teach, reiterate, or reinforce a specific aspect related to the failure.
Establish a program or service	Establish a program to aid a process, such as a record-keeping program.
Establish an independent and transparent supervisory agency	Establish an agency that acts as a watchdog for an aspect of the failure.
Establish more checks in the system	Put more checks in the system, for example a supervisor's signoff, such that work cannot continue without conducting the check.

Table 2.5 continued

Give supervisor more capacity for oversight	Provide supervisors with the power to enforce the rules to which systems are required to adhere.
Identify weak areas	Assess what aspects of the system may be neglected.
Improve efficiency in critical tasks	Improve how a task is done, for example by eliminating steps in a procedure, providing better equipment, or making software assistance to operators more logical.
Increase resources	Provide more aspects like people, money, or equipment, to an aspect of the system.
Involve stakeholders in decision-making	Involve more stakeholders to provide additional points of view that were previously lacking.
Keep up with current technologies	Improve technological aspects of the system like outdated computer systems, or emergency systems.
Make instructions more clear	Improve instructional aspects of the system, such as procedures, job descriptions, employee roles, or any other type of instruction to be clearer.
Make regulations more strict	Improve regulations to make the standards to which the system is held to be more stringent.
Review decision-making logic	Instead of incrementally making small changes to a system, rather, for example, change how aspects of the system are addressed or review the system from a high-level perspective.
Track compliance to an objective standard	Hold system activities to applicable standards, such as ensuring drawings follow a template or having every employee complete the same training.

Last, we linked the causes from the actor-causal action-object codes to the recommendation codes. We linked only those recommendations that we could reasonably infer corresponded to the causes we identified. For example, if accident investigators found problems with the way a subsytem was tested, and then made recommendations for improving that subsystem test, we inferred that those two items were linked. When can we not reasonably infer that a recommendation is applicable to a cause we identified? In the Alaska Airlines flight 261 aircraft crash, maintenance personnel consistently did not lubricate the jackscrew assembly in the horizontal stabilizer properly, eventually leading to the component failing and causing the aircraft to crash. The accident investigators made many recommendations to improve the maintenance process at the airline, but they also suggested that Alaska Airlines establish the lubrication procedure as a required inspection item that must have an inspector’s signoff. We thought that this recommendation was more appropriately applied to poor inspection practices at the company, rather than poor maintenance practices. Figure 2.8 displays the recommendation code distribution for *managed risk poorly*. Overall, we did not find recommendations for 30% of the accident causes.

Managed Risk Poorly

102 **Total instances identified**	=	**9** **Instances clearly linked to recommendations**	+	**44** **Instances linked to recommendations by inference**	+	**49** **Instances not linked to recommendations**

Figure 2.8: Recommendation code distribution for *managed risk poorly*

2.5 Cause Networks and the Cause-Recommendation Network

We have identified over 900 specific examples of failure causes and 600 specific examples of remedial actions. Here, we develop a graphical network to facilitate navigation of the results.

The cause network is based on the cause presence and the probabilities of finding pairs of causes in a given accident or project failure. Table 2.6 shows the intersectional probabilities $P\left(cause_i \cap cause_j\right)$ for “failed to consider human factor” ($cause_i$) and all the other causes for both accidents and project failures. For example, *failed to supervise* occurred together with *failed to consider human factor* in 21% of project failures, and 37% of accidents.

Table 2.6: Intersectional probability of *failed to consider human factor* with the other causes

$Cause_j$	$P(failed\ to\ consider\ human\ factor \cap cause_j)$	
	Project Failure	**Accident**
Failed to supervise	21%	37%
Failed to provide resources	18%	23%
Failed to consider design aspect	27%	37%
Lost tacit knowledge when employee departed	6%	3%
Lacked experience	15%	20%
Used inadequate justification	6%	23%
Subjected to inadequate reviews	3%	10%
Kept poor records	6%	13%
Failed to form a contingency plan	12%	10%
Inadequately communicated	18%	10%
Conducted poor requirements engineering	15%	13%
Failed to inspect	9%	17%
Violated regulations	3%	20%
Managed risk poorly	18%	30%
Subjected to inadequate testing	12%	7%
Violated procedures	6%	23%
Did not allow aspect to stabilize	15%	3%
Did not learn from failure	3%	27%
Conducted maintenance poorly	0%	27%
Created inadequate procedures	6%	30%
Enforced inadequate regulations	0%	27%
Failed to train	0%	40%

These percentages are difficult to interpret in a table format, as we presented in Table 2.6. Thus, we plotted the intersectional probabilities of causes for accidents and for project failures as undirected graphs, as shown in Figure 2.9 and Figure 2.11. The nodes represent the causes, and the links represent the cause intersectional probabilities. Heavy links indicate high intersectional probabilities, thin links the opposite. Large nodes indicate a high cause presence, small nodes the opposite. The plot is laid out like a force model, so that nodes that are linked attract each other, and nodes that are not linked repel each other.

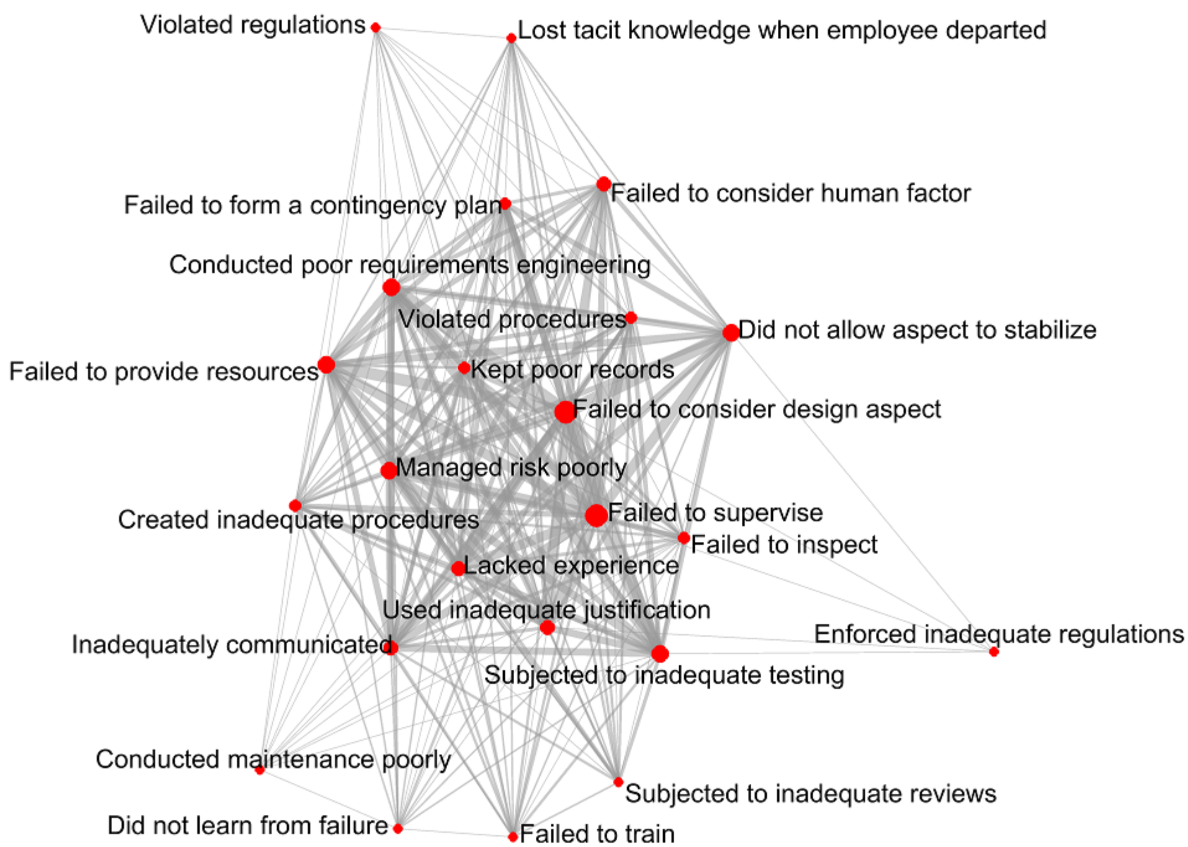


Figure 2.9: Project failure cause intersection likelihood

These graphs are quite dense, however, so Figure 2.10 shows an illustrative example of the information for the project failure graph using a subset of the causes. We plotted six of the twenty-three causes on a skeleton of the project failure undirected graph (*failed to consider human factor, failed to consider design aspect, failed to supervise, created inadequate procedures, enforced inadequate regulations,* and *did not learn from failure*). For this illustrative example, we selected causes that have high presence and many interconnections (*failed to consider design aspect* and *failed to supervise*), causes that have low presence and few interconnections (*enforced inadequate regulations* and *did not learn from failure*) and causes between those extremes (*failed to consider human factor* and *created inadequate procedures*). Figure 2.10 shows that causes with few interconnections are on the outside of the graph, while causes with many interconnections are concentrated on the inside of the graph (a "force" layout). The connections between causes with high presence (larger nodes) are thicker than the connections from one cause to another with low presence (smaller nodes) because if we did not identify a cause in many failures it will make sense that we did not identify that causes with other causes very often.

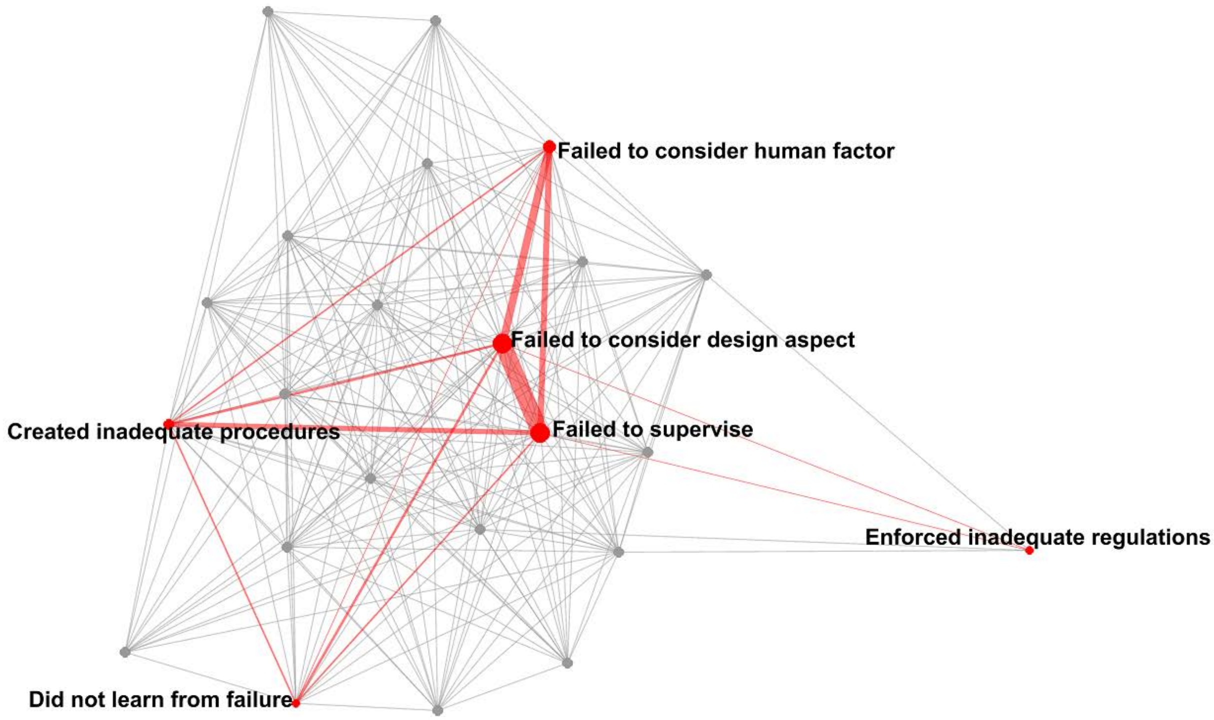


Figure 2.10: Project failure cause graph illustrative example

In project failures (Figure 2.9) the eight causes with low presence (<20%), such as *enforced inadequate regulations*, are all outlying nodes with thin connections. Similarly, the five causes with low presence in accidents (Figure 2.11), such as *did not allow aspect to stabilize*, are all outlying nodes with thin connections. The two causes with high presence (>70%) in project failures (*failed to consider design aspect* and *failed to supervise*) are both internal nodes in with many thick connections. Similarly, the three causes with high presence in accidents, such as *managed risk poorly*, are also internal nodes with many thick connections. Figure 2.9 has more outlying nodes, with thinner connections on average than Figure 2.11. The causes in project failures generally have lower presence values than causes in accidents, which means there are fewer opportunities to be connected to the other causes.

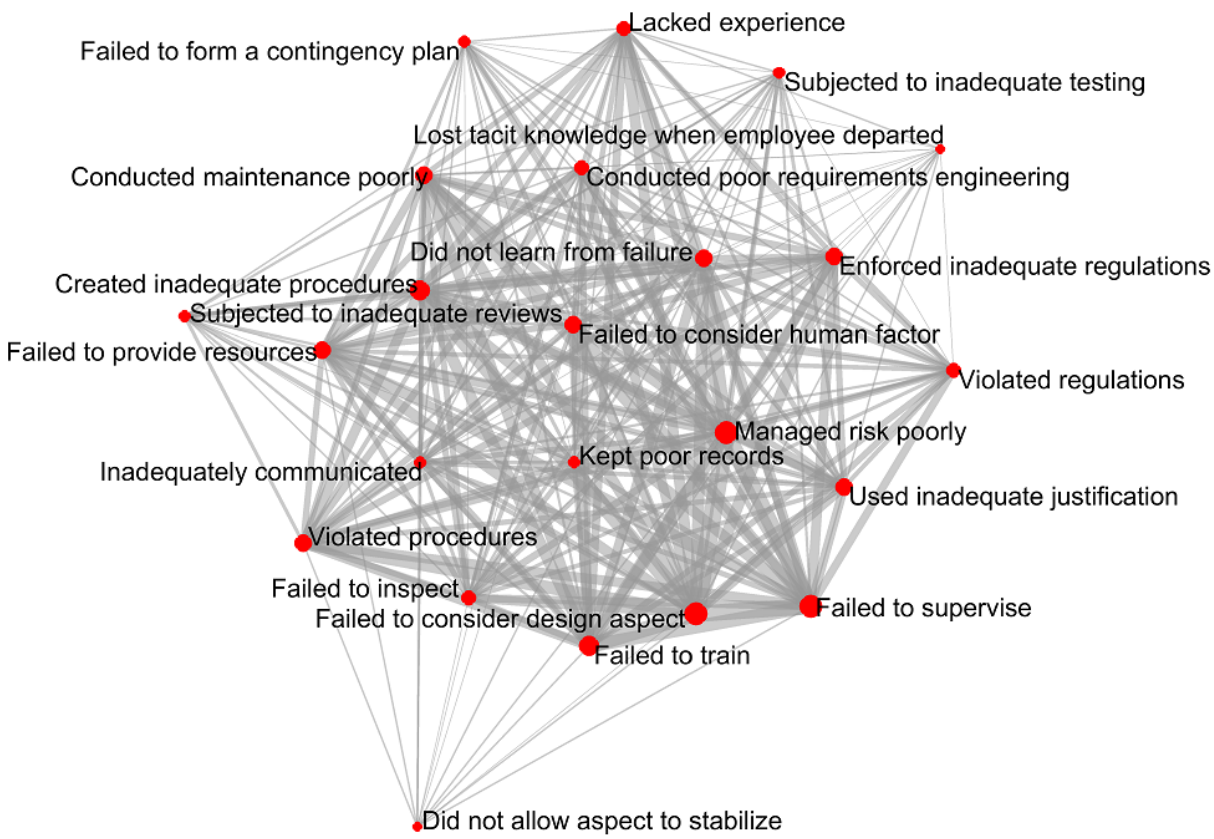


Figure 2.11: Accident cause intersection likelihood

Next, we built a cause-recommendation network using the links we identified between the causes and the recommendation codes. In Figure 2.12, the red nodes are causes, and the blue nodes are recommendations. For clarity, we have omitted the cause-cause links. Like the cause networks, nodes with many connections repel nodes with few connections. Thin links indicate that the cause and recommendation were connected only one or two times; heavy links the opposite, with the thickest line indicating 49 connections between *managed risk poorly* and *no recommendation* (see Figure 2.8). Some causes only have a few recommendations; this situation occurs when causes are quite specific and also have quite specific recommendations. For example, a frequent recommendation for *subjected to inadequate testing* is *develop a more comprehensive and rigorous test* (that is, a frequently suggested solution to inadequate testing is adequate testing!). Other causes are more ambiguous and are thus covered by a wider range of recommendations. Such causes include *failed to supervise*, which is covered by recommendations like *conduct random and independent evaluations* and *develop specialized training*.

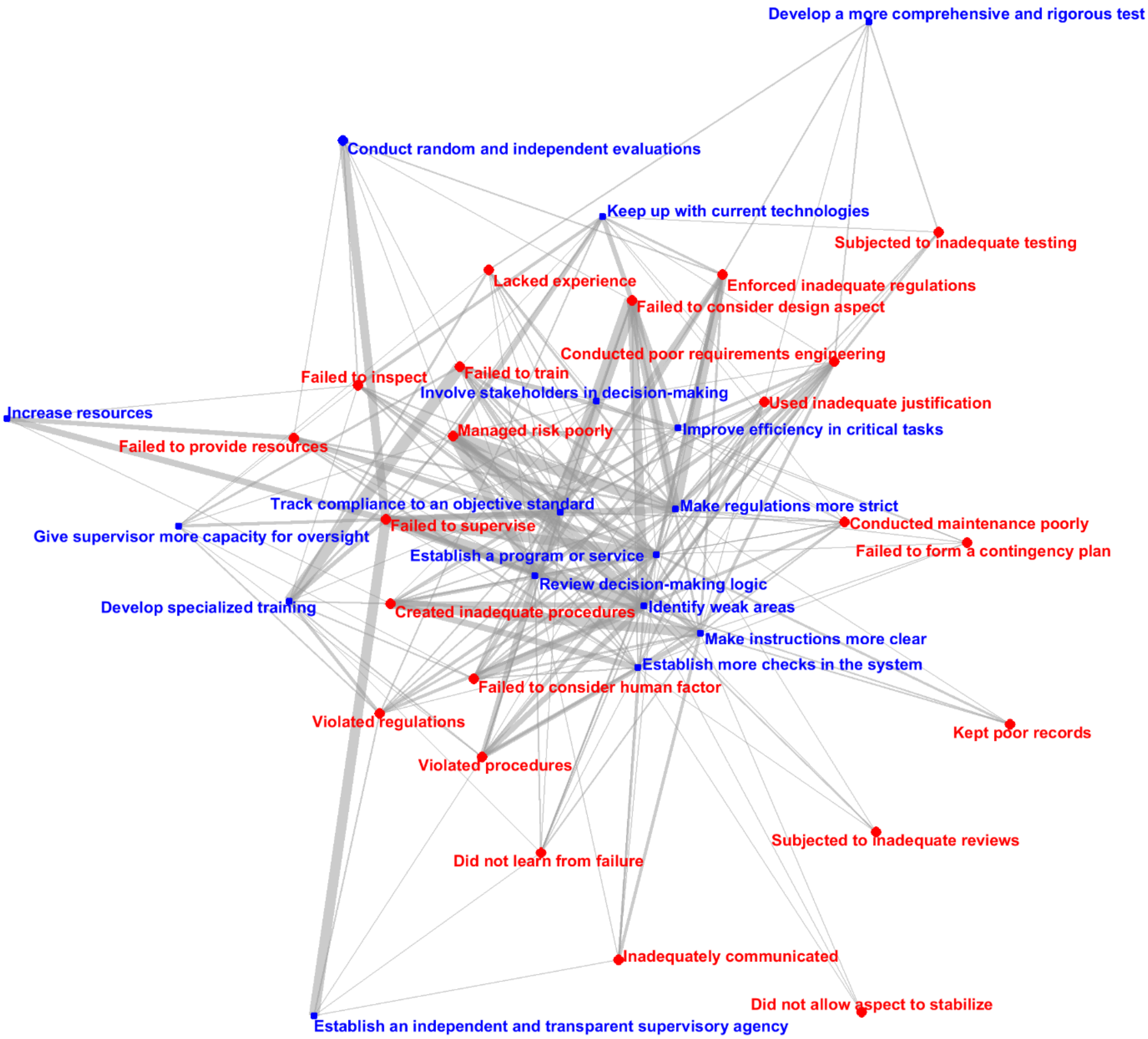


Figure 2.12: Cause-recommendation network

The cause-recommendation network is similar in construction and representation to a Bayesian Network, in which the direct influence of one feature to another feature is represented using conditional probabilities [Neapolitan, 2003]. In a Bayesian Network, however, each node is shown as a result of another, in a directed graphical form. Our network is more similar to a Markov network (or Markov random field), which has undirected edges between nodes [Ben-Gal, 2007; Neapolitan, 2003]. Since we did not construct our network using true conditional probability calculations, it is not a true Markov network.

2.6 Analysis of Failures at Large-Scale Aerospace Companies

Our study of past systems engineering failures is based on information reported by third party investigators; regulators, journalists, and accident investigators (for a more detailed discussion on our sources refer to Section 2.2.1). How do internal company investigations compare to the investigations undertaken and reported by third parties we studied? How do problems on company projects as identified by systems engineers contribute to these failures, and how do these problems compare to the ones identified by non-systems engineers from our study?

We identified three large-scale aerospace companies who expressed interest in giving us a more in-depth view of their systems engineering processes than is usually recorded in failure reports. Two companies followed through and agreed to work with us. These companies all employ tens of thousands of people worldwide, and each ranks within the top defense contractors based on their defense revenue [DefenseNews, 2018].

“Company A” preferred interacting with us using a paper format, where we asked them written questions and they provided a written response. Section 2.6.1 describes the questionnaire we submitted to Company A and Section 2.6.2 describes and analyzes the response we received from the company.

“Company B” preferred interacting with us in-person in semi-structured interviews. Section 2.6.3 describes the questions that comprised the semi-structured interview, and Section 2.6.4 describes and analyzes the responses we collected from 8 systems engineers at Company B.

The data from the two companies in general aligns with the findings from our analysis of accident and project failure reports. Company A provided analysis on 5 failures they encountered, where they systematically went over each contributing factor and coded their findings, which is similar to the way we analyzed past systems engineering failures (see Section 2.2.2). The interviews we conducted with Company B systems engineers allowed us to tap into their systems engineering expertise in a much more detailed way. We were able to gain insights into their day-to-day activities, such as what tools they use, as well as their observations on the systems engineering industry based on their years of experience. The systems engineers described how their industry has changed over time and what the potential gaps in systems engineering education may be based on what they have observed newly hired systems engineers struggling with. Our discussions in Sections 2.6.2 and 2.6.4 indicate that further survey of systems engineering failures at companies will provide insights that will further enhance the industry's study of systems engineering failures because these surveys capture systems engineers' perspectives that are absent from readily available information on systems engineering failures.

2.6.1 Paper Questionnaire Development

The purpose of this survey was to ask systems engineers about how problems with systems engineering specifically may contribute to different types of failures, such as accidents, cost overruns, and quality concerns[3]. We wanted to determine whether (1) systems engineers believe certain failures are more prevalent at their company than what studies on systems engineering failures indicate, and (2) systems engineers believe that problems identified in systems engineering literature contribute differently to the failures they experienced or in more specific ways that our study did not uncover. For example, a systems engineer may have experienced "not using lessons learned" impacting project cost because manufacturing mistakes were not being corrected at their company and parts needed to be scrapped frequently.

The survey begins by asking the respondent about their background, both generally as a systems engineer and also specifically while working at their company. Then, to prompt the respondent to think about systems engineering failures at their company, the survey asks the respondent to estimate how many projects they are involved in or have observed at their company that have experienced each type of failure. This is meant to prompt the respondent to give a broad view of project performance at their company. To determine the role of systems engineering specifically in these failures at their company, the survey also asks the respondent to estimate how many of these failures have been related to systems engineering (i.e., how much the systems engineering or lack thereof on a project contributed to a failure). The survey then asks the respondent how severely/frequently 20 "issues" (such as "not using lessons learned") have contributed to each

[3] We wrote this paper survey during the early stages of this research, before conducting our analysis of past systems engineering failures. We thus cannot directly compare the problems we asked the systems engineers to elaborate on and the problems we identified in our study of past systems engineering failures.

failure using a risk matrix representation. The survey concludes with a free-form response asking the respondent to describe what most important change is needed in systems engineering, implicitly asking the respondent how systems engineering itself could change to prevent these failures from occurring. Appendix A.1 contains a blank copy of the survey.

We wrote the survey around 5 types of systems engineering failures: accidents/incidents, quality concerns, cancellations, performance gaps, and delay and cost overruns. Table 2.7 contains the information we gave on each failure in the paper survey[4].

Table 2.7: Systems engineering failure descriptions given in paper survey

Systems Engineering Failure	**Additional Information**
Accident/Incident	On 23 September 1999, communication with the Mars Climate Orbiter was lost as the spacecraft went into orbital insertion. The insertion failed because of a units mismatch between NASA and Lockheed. The spacecraft encountered the Martian at an improperly low altitude, causing it to incorrectly enter the upper atmosphere and disintegrate.
Quality Concerns	Toyota issued a recall in January 2010 due to possible mechanical sticking of the accelerator pedal causing unintended acceleration.
Cancellation	The X-33 was cancelled in 2001 after a long series of problems with flight stability and excess weight.
Performance gap	Iridium filed for bankruptcy in 1999 after it failed to garner enough subscribers. This failure was due in part to poor phone coverage. Because the technology depended on line-of-sight between phone antennas and the orbiting satellites, the phones did not work inside moving cars, inside buildings, and in many urban areas. Iridium was subsequently reborn, but at a much smaller scale than originally envisioned.
Delay and cost overrun	The Boston Big Dig was originally scheduled to be completed in 1998 at an estimated cost of $2.8 billion (1982 dollars). The project was plagued with technical, scheduling, cost, and even criminal problems. It was eventually completed in in December 2007, at about 190% of the originally planned budget (over $14.6 billion in 2006 dollars).

We asked respondents to consider issues that contribute to systems engineering failures and to use a risk matrix to rate how severely and frequently twenty possible problems identified in the systems engineering literature contribute to these failures[5] (e.g., how severely/frequently did "ineffective

---

[4] The information for "accident/incident" describes the Mars Climate Orbiter loss, which we defined in in Section 2.2 as a type of project failure ("failure to meet mission objectives"). This also fits under the description of an accident we defined, however, as it was an event that directly resulted in damage to property. We used this example specifically for ease of understanding by the respondents.

[5] Note that since we conducted this study in parallel with the report analysis, the "problems" we use here do not match one-to-one with the "causes" in the cause-recommendation network.

risk management procedures" contribute to "performance gaps"?). Table 2.8 describes each problem and shows the source of each problem.

Table 2.8: Glossary of problems that contribute to undesired events

#	Problem	Additional information	Source
1	Weakness in otherwise good processes	A single weakness can devastate a manufacturing process, for example.	[Newman, 2001]
2	Loss of company knowledge due to employee retirement	Employees are a wealth of project history and engineering knowledge, which can be lost when an employee retires.	
3	Inability to hire enough systems engineers	Systems engineers can be useful in managing complex projects and processes.	[NDIA, 2008]
4	Not using "lessons learned"	This issue can be a problem in large or far-reaching companies where projects or departments do not necessarily communicate between each other. If a solution is found to a common problem it may not be communicated to other places the same problem is present.	[Newman, 2001]
5	Process inadequate for complex systems	A process will not be as effective for systems of different complexities	[Bar-Yam, 2003]
6	Inadequate planning in the early stages	This issue can be detrimental to schedule, for instance, if there is optimistic planning in the early stages.	[NDIA, 2008]
7	Problems with staff, training, or expertise	It is desirable for a workforce to be completely versed in the products or systems and have adequate training.	
8	Lack of rigor	A generally "lax" company can have trouble with a variety of issues, like enforcing safety protocols.	[Winter, 2007]
9	Ineffective resource allocation	Resources must be allocated appropriately, especially in a company with multiple projects at varying degrees of completion.	
10	Inadequate requirements engineering	Requirements must be defined in order to know how to design and test a product, for example.	[NDIA, 2008]
11	Inadequate knowledge transfer	For example, new engineers often graduate from school with only the basic tools to do engineering work, and they must be trained by their coworkers to do useful work on a project.	[Winter, 2007]
12	Lack of employee loyalty	Employees must care about the companies' overall success and have a desire to do good work.	
13	Issues with reward and incentive structures	Incentives can for example prioritize schedule over quality, which is often more difficult to quantify.	
14	Ineffective risk management procedures	Risks are present at every company, but they become dangerous if they are underestimated, not identified, or not addressed properly.	[Nowinski & Kohler, 2007]
15	Ineffective team building	Engineers generally work in teams.	

Table 2.8 continued

16	Acquisition management failures	Improper acquisition management can result in a product that does not meet the desired needs, is over-budget or behind schedule, or does not meet the test standards.	[Smith, 2007]
17	Ineffective government/industry teaming	Often there are a lot of administrative obstacles a company must work through in order to effectively do work with the government.	
18	Real time efficiency decisions that led to later problems	Employees can cut corners to save time, but this could be detrimental to aspects like safety and performance.	[Thomas, 2007]
19	Dysfunctional feedback across the system lifecycle	It is important to know about issues like an incorrectly interpreted requirement as early as possible so that work does not have to be redone.	[Triantis et al., 2009]
20	Using process rather than thinking and being accountable	Work instructions, for example, can be very useful when assembling a product. However, they are not useful for new and complex products.	[Slegers et al., 2011]

Figure 2.13 shows the format of the risk matrix that we asked the respondents to use.

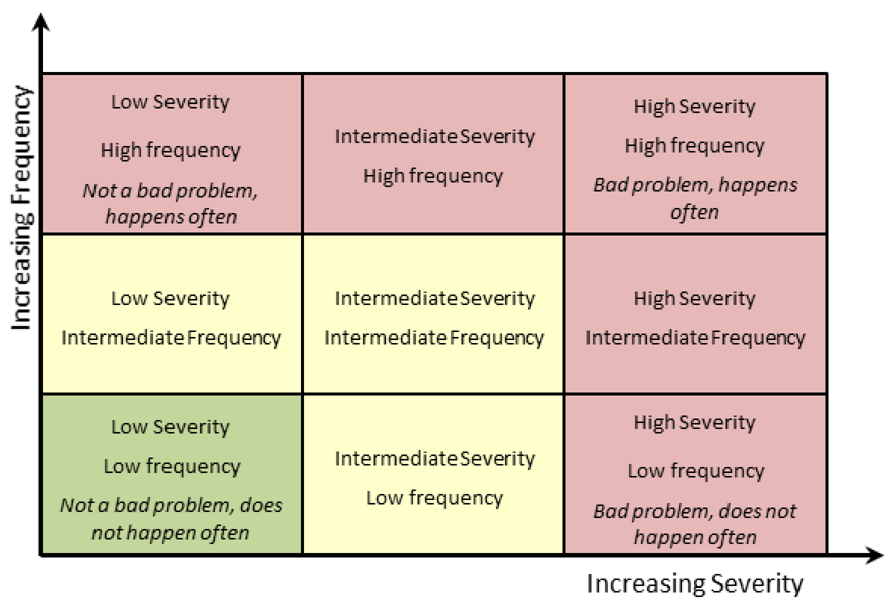


Figure 2.13: Survey risk matrix explanation

2.6.2 Company A Response

Although we anticipated and hoped for multiple independent responses, Company A returned a single filled-out copy of this survey to us that had been completed by multiple systems engineers working together. This "committee" method of filling out the survey may have made it easier for Company A to ensure that they did not inadvertently release information they did not want to. Company A also provided supplemental information about failures that occurred at their company. Table 2.9 contains the estimates from the Company A respondents on how many projects they have worked on that experienced the five types of failures (left column). We also asked the respondents to estimate the role of systems engineering specifically in these failures at their company, by estimating how many of these failures have been related to systems engineering (i.e., how much the systems engineering or lack thereof on a project contributed to a failure; right column).

Table 2.9: Overall project performance

Type of event	**Percentage of projects suffering from event**	**Percentage of events related to systems engineering**
Accident/incident	33%	33%
Quality concerns	50%	33%
Cancellation	10%	10%
Performance gap	10%	10%
Delay	10%	10%
Cost overrun	33%	20%
No significant failures	<10%	<10%

The respondents indicated that *quality concerns* plague their engineering projects the most, with *accident/incident* and *cost overruns* as the second-highest frequency events. When estimating how these failures related to systems engineering efforts, the respondents indicated that systems engineering (or lack thereof) played a proportional role in these failures, with *quality concerns*, *cost overruns*, and *accidents/incidents* having the highest frequencies.

Fewer than 10% of their engineering projects suffer from none of these events, which is similar to the information we found on engineering projects overall, especially the statistics on US defense programs (recall that Charette [2008] stated that out of 72 major U.S. defense programs in progress in 2008, only eleven of them were on time, on budget, and met performance criteria and that problems have only gotten worse since then [Lineberger & Hussain, 2016].)).

Table 2.10 contains the respondents' estimates on how severely and frequently certain problems contributed to these failures.

Table 2.10: Risk matrix data from Company A survey. each box indicates how the representative indicated the problem's severity and frequency using the risk matrix; whether the problem was "high" for severity and/or frequency (red), "intermediate" for both severity and frequency (yellow), "low" for both severity and frequency (green), or if the representative did not indicate how frequent or severe the problem was (gray).

Issue	Accidents and incidents	Quality concerns	Cancell-ations	Performance gaps	Delays	Cost overruns
Weakness in otherwise good processes		Inter-mediate		Intermediate	Low frequency and severity	Low frequency and severity
Loss of company knowledge due to employee retirement		Inter-mediate		Intermediate	Inter-mediate	Low frequency and severity
Inability to hire enough systems engineers		Inter-mediate			Inter-mediate	Inter-mediate
Not using "lessons learned"	High Severity	High Frequency		Intermediate	Inter-mediate	Inter-mediate
Process inadequate for complex systems		Inter-mediate		Intermediate	Inter-mediate	Inter-mediate
Inadequate planning in the early stages		Inter-mediate		Intermediate	High Frequency	High Frequency
Problems with staff, training, or expertise		Inter-mediate		High Frequency	Inter-mediate	High Frequency
Lack of rigor		High Frequency		Intermediate	Low frequency and severity	High Frequency
Ineffective resource allocation		Inter-mediate		Intermediate	High Frequency	Inter-mediate
Inadequate requirements engineering		High Frequency		High Frequency	Inter-mediate	High Frequency
Inadequate knowledge transfer	High Severity	Inter-mediate		Intermediate	Inter-mediate	Inter-mediate

Table 2.10 continued

Lack of employee loyalty						
Issues with reward and incentive structures						
Ineffective risk management procedures	High Severity	High Frequency		High Frequency	High Frequency	High Frequency
Ineffective team building		Inter-mediate			Inter-mediate	Inter-mediate
Acquisition management failures	High Severity	Inter-mediate		Intermediate	High Frequency	High Frequency
Ineffective government/industry teaming		Inter-mediate	High Severity		Inter-mediate	High Frequency
Real time efficiency decisions that led to later problems	High Severity	Inter-mediate		High Frequency	High Frequency	High Frequency
Dysfunctional feedback across the system lifecycle		Inter-mediate		High Frequency	High Frequency	High Frequency
Using process rather than thinking and being accountable		Inter-mediate		Intermediate	Inter-mediate	High Frequency

None of the Company A representatives marked any of these issues as both high severity and high frequency in any failures. If they indicated an issue was in the “high” category, it was for either severity or frequency.

Cost overruns had the most high-frequency issues associated with it. The representatives said that 9 of the issues were high-frequency for this failure. Delays, quality concerns, and cost overruns all had the most issues associated with them, although at varying degrees. The representatives indicated that 18 out of 20 issues were associated with these failures (and the remaining two issues were not associated with any failures at all).

Cancellations had the fewest issues associated with it (high, intermediate, or low). The representatives said the only issue that contributes to this failure is ineffective government/industry teaming. This makes sense because this company works primarily on defense contracts, so a cancellation would be at the discretion of the government.

While accidents and incidents had few issues associated with it, the Company A representatives marked the issues that were as all high-severity. The representatives likely take accidents very seriously and consider issues contributing to them as a high priority.

In addition to answering the survey we provided, Company A provided supplemental information on failures they experienced. The respondents provided the information on these failures in a drill-down format, where they reported on how they discovered a problem, then described how they investigated the problem and what they found. This data provided insights into systems engineering failures at Company A that the survey did not capture, as we discuss next.

In the introduction to this section we discussed how the purpose of interviewing systems engineers at large-scale aerospace companies was to determine how internal company investigations compare to the investigations undertaken and reported by third parties we studied, and how problems in systems engineering as identified by systems engineers contribute to these failures. To answer these questions, we compare the paper survey responses and the supplemental data Company A provided to our study of past systems engineering failures. To do this comparison we considered three questions:

1. Can we code the supplemental data Company A provided in a similar way to our study of past systems failures and compare the results?
2. What causes are present in:
    a. The supplemental data that are also present in the data we collected on past failures and do they appear in a similar frequency?
    b. The paper survey that are also present in the data we collected on past failures and do they appear in a similar frequency?
3. Are the problems the representatives identified in the survey also present in the data they provided on failures?

Can we code the supplemental data Company A provided in a similar way to our study of past systems failures and compare the results? For each failure, Company A provided narratives that described the problem they encountered, steps they took to investigate the cause(s) of the problem, and a "root cause" that they assigned a code to. Since the format of the narratives appeared similar to the sources we found on past systems engineering failures, we were able to code the data in a manner similar to the method we used for our analysis of past systems engineering failures (see Section 2.2.2) and compare the causes we identified to the codes that Company A assigned each root cause. Note that in our research we consider all causes as contributors to each failure and we do not place emphasis on any single "root cause". Therefore, we focused on the narratives and did not assign any codes to the root causes. Table 2.11 shows an excerpt from one of these failures, the code we assigned to the findings, as well as the code the company assigned to the root cause they identified.

Table 2.11: Excerpts of Company A data coding

<table>
<tr><th>Failure and Description</th><th>Finding</th><th>Researcher-assigned Code</th></tr>
<tr><td rowspan="4">Failure 1: Oil Temperature Sensing System: A high oil temperature was detected, and the pilot reduced power but the oil temperature did not drop. The oil temperature sensor had failed and caused an in-range high reading of oil temperature.</td><td>"the design incorporated two temperature sensing elements, but the temperature at which the system sensed that the sensor had failed and switched to the alternative sensor was 300ºC. By contrast, the temperature at which the pilot would initiate the actions that resulted in the commanded shut-down was 127ºC."</td><td>Designers conducted poor requirements engineering (performance requirements).</td></tr>
<tr><td>"No analysis had been performed to understand the relative probabilities of different failure mechanisms (out-of-range low, in-range low, in-range high and out-of-range high) to inform the system design."</td><td>Designers managed risk poorly (risk likelihood).</td></tr>
<tr><td>Company A-Assigned Root Cause</td><td>Company A-assigned Code</td></tr>
<tr><td>"The oil temperature sensing system design did not take advantage of the dual sensors and as a result was vulnerable to in-range high failures of sensor 1."</td><td>"Lack of Appropriate Risk Management"</td></tr>
</table>

What causes are present in the supplemental data that are also present in the data we collected on past failures and do they appear in a similar frequency? Many of the codes the respondent assigned to the findings were similar to the codes we developed in our analysis of past systems engineering failures, such as lack of risk management (we identified as "managed risk poorly") and inadequate verification (we identified as "subjected to inadequate testing"). Table 2.12 shows the number of instances of each cause we identified in the Company A data, as well as the codes the respondent identified in their own data, and how they best map to our codes.

Table 2.12: Company A cause mapping

Researcher-Assigned Code	**Company A-Assigned Codes**	**Number of Instances in Company A Data**	**Number of failures code was identified in (out of 5)**
Conducted poor requirements engineering	The requirements set is incomplete or not being effectively managed through the project lifecycle	7	3
Failed to consider design aspect	Configuration Management Issue Inadequate/Missing Trade Study	5	2
Subjected to inadequate testing	Inadequate Verification, either by test or analysis	4	3
Used inadequate justification	Incorrect Assumption	3	2
Created inadequate procedures	The process is deficient	1	1
Kept poor records		1	1
Lacked experience		1	1
Managed risk poorly	Lack of appropriate risk management	1	1
Violated procedures		1	1

To visually compare the results of the Company A data with the results of our study of past systems engineering failures, we plotted the Company A data in an undirected graph format, similar to how we display the project failure causes in Section 2.5. In that section, Figure 2.9 is an undirected graph that represents the project failure causes, where the node size is the presence (a percentage that represents in how many project failures we identified a certain cause) and the line weight indicates how frequently we identified each cause together in the same project failure. Figure 2.14 displays the Company A data (red lines/nodes) plotted on a skeleton of Figure 2.9 (gray lines/nodes). The gray lines/nodes indicate a "neutral" background against which to compare the Company A data. The blue nodes and lines are specific to the Company A data in that bigger nodes indicate that we identified a certain cause in relatively many Company A failure narratives (e.g. we identified "conducted poor requirements engineering" in 3 of 5 failure narratives) and heavier connecting lines indicate that we identified two causes together in relatively many Company A failure narratives (e.g. we identified “conducted poor requirements engineering” and “used inadequate justification” together in 2 of 5 failure narratives).

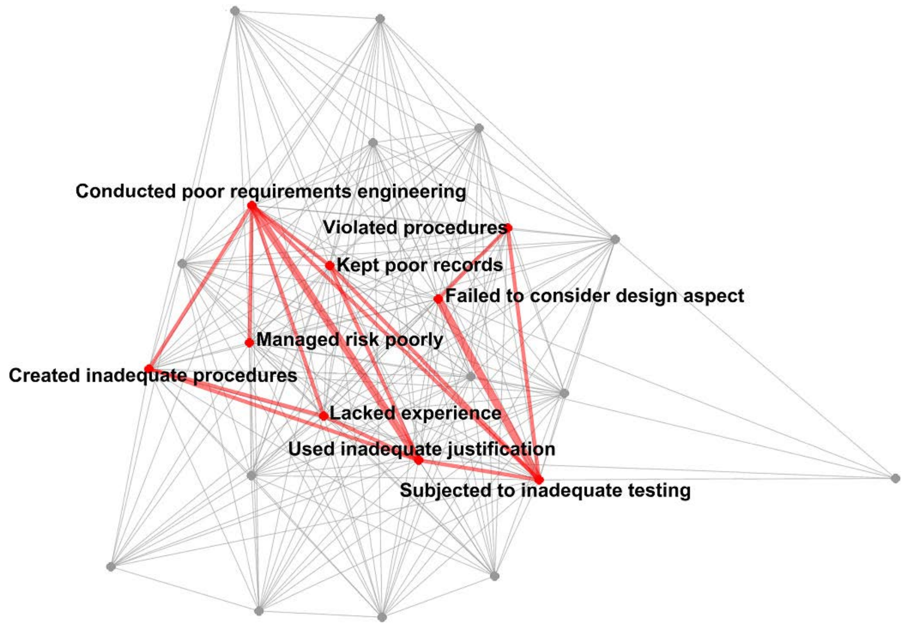


Figure 2.14: Company A cause intersection

Figure 2.14 shows how the scale of the Company A data compares to our study of previous engineering failures. We identified only 9 of our 23 causes within this data, and only a few of those causes within the same failure. Of the causes we identified in both sets of data, many have similarly-scaled presences. “High presence” causes (we identified in 2 or more failures):

- Failed to consider design aspect: high presence in project failures (85%)
- Conducted poor requirements engineering: high presence in project failures (52%)
- Subjected to inadequate testing: present in some project failures (45%)
- Used inadequate justification: present in some project failures (42%)

“Low presence” causes (we identified in 1 failure):

- Created inadequate procedures: present in few project failures (18%)
- Kept poor records: present in few project failures (21%)
- Violated procedures: present in few project failures (21%)
- Lacked experience: present in some project failures (42%).
- Managed risk poorly: present in some project failures (48%)

Many causes were not present at all in the company data. The respondents notably did not indicate "people problems" as causes of failures in the data they provided, such as problems relating to managing/supervising, training, communication, human factors, or team members' experience. This could indicate a difference in the priorities of company internal failure investigations and publicly-available failure investigations. We do not think that companies disregard these "people problems" as factors in the problems their projects experience, but it is of note that they are willing to publicly share "engineering problems" but not "people problems" on their projects.

What causes are present in the paper survey that are also present in the data we collected on past failures and do they appear in a similar frequency? First, we compare the scale (severity + frequency) of the problems the respondents identified with the frequency of similar causes (or subsets of causes) we identified in our study of past systems engineering failures. Some of the responses to this survey are directly comparable to findings from our study of systems engineering accidents and project failures:

- "Acquisition management failures": the respondents identified this problem as high-severity in accidents, and high-frequency in delays and cost overruns. In our study of accidents and project failures, we also found that management "used inadequate justification" for acquisitions specifically for 4 project failures (12%) and for 1 accident (0.3%), so this was not an insignificant problem in project failures we studied.
- "Not using lessons learned": the respondents identified this problem as high-severity in accidents and high-frequency in quality concerns. This is similar to our study in which "did not learn from failure" had a higher presence in the accidents we studied than the project failures we studied.
- "Ineffective risk management procedures": the respondents identified this problem as a high severity/frequency problem in every failure except cancellations. This is similar to our finding that "managed risk poorly" had a high presence in both the accidents and project failures we studied. The industry representatives specified that ineffective risk management had high **severity** in accidents but high **frequency** in project failures, but our analysis indicated that poor risk management has a higher presence in the accidents we studied than the project failures we studied. In Section 2.6.1 we discussed that we defined "accident/incident" for the paper survey as an event that directly leads to property damage,

so it makes sense that an industry representative would view an accident/incident being intrinsically more severe in consequence and will thus rate problems with risk management as contributing more severely to this event than for another event.

- "Loss of company knowledge due to employee retirement": the respondents identified this problem as a low or intermediate severity/frequency problem in project failures, which is consistent with our finding that "lost tacit knowledge when employee departed" has a low presence in the accidents and project failures we studied.

Some of the findings from this survey that did not match the findings from our study of systems engineering accidents and project failures:

- "Inadequate requirements engineering": the respondents identified this problem as a high-frequency problem exclusive to quality concerns, performance gaps, and cost overruns. We found that "conducted poor requirements engineering" was a nearly consistent problem across the accidents and project failures we studied. This difference could be because our definition of "requirements engineering" is broader than what a company may define the term as. Our definition included "laying out the needs, attributes, capabilities, characteristics, or qualities of the system well" and did not exclusively apply to findings that specified problems with "requirements", since the accident and failure reports were likely not written by systems engineers familiar with this coded language. For example, in the Westray Mine disaster, "the ventilation system in the North Mains and Southeast sections of the mine was haphazard, reflecting little or no planning" [Richard, 1997]. This finding relates specifically to planning the mine layout and design, which fits within our definition of requirements engineering. However, since the finding does not specifically use the word "requirement" or relate directly to "requirements management", a company may not also code this finding in a similar way.
- "Ineffective resource allocation": the respondents identified this problem as a high-frequency problem in one type of failure: delays (though not in "cost overruns"). However, we found that "failed to provide resources" had very high presence in the accidents and the project failures we studied. This could also be the result of a disconnect between our definitions of "resources". We consider "resources" to contain more than just funding; our

definition includes employees, technology, and equipment, to list a subset. The respondent may have had a different definition of this term and thus applied it to failures differently.

The respondents did not identify two issues as contributing to any failures: “lack of employee loyalty” and “issues with reward incentives”. However, we found problems with “work design and organization human factors” (which is how we classified problems with “how workers work”, such as encouraging workers to work long, fatigue-inducing hours) in 5 project failures and 11 accidents. Possible explanations for this difference include: (1) this problem does not happen at this company, (2) the respondents have never been exposed to this problem at their company, or (3) the respondents have seen this problem at their company and they did not wish to identify it. With the limited data we have, we cannot determine which explanation(s) is/are true.

Lastly, are the problems the representatives identified in the survey also present in the data they provided on failures? To compare the Company A paper survey results to the Company A failure narratives, we identified what type of failure each narrative corresponded to and coded the findings from each narrative into “issues” from the paper survey. Four of the Company A failure narratives discuss problems with parts breaking and causing wider system failures, so we identified those narratives as describing “quality concerns”. One of the Company A failure narratives discusses performance problems for a subsystem, so we identified that narrative as describing a “performance gap”. Table 2.13 contains the results of the process of grouping each finding into issues from the paper survey and compares these results to the risk matrix data from the paper survey for quality concerns and performance gaps. The numbers for the narrative columns indicate how many of the failures had findings that coded into each issue (e.g. 3 out of 4 quality concerns had findings that we coded as “ineffective risk management procedures”). The gray boxes in this table indicate that we do not have data for that issue (e.g. we did not code any findings from quality concerns as “lack of employee loyalty”).

Table 2.13: Risk matrix data from Company A survey compared to Company A failure narratives

Issue	Quality concerns (paper survey)	Quality concerns (4 narratives)	Performance gaps (paper survey)	Performance gaps (1 narrative)
Weakness in otherwise good processes	Intermediate	1	Intermediate	1
Loss of company knowledge due to employee retirement	Intermediate		Intermediate	
Inability to hire enough systems engineers	Intermediate			
Not using "lessons learned"	High Frequency		Intermediate	
Process inadequate for complex systems	Intermediate		Intermediate	1
Inadequate planning in the early stages	Intermediate	2	Intermediate	
Problems with staff, training, or expertise	Intermediate	1	High Frequency	
Lack of rigor	High Frequency	2	Intermediate	1
Ineffective resource allocation	Intermediate		Intermediate	
Inadequate requirements engineering	High Frequency	1	High Frequency	1
Inadequate knowledge transfer	Intermediate		Intermediate	
Lack of employee loyalty				
Issues with reward and incentive structures				
Ineffective risk management procedures	High Frequency	3	High Frequency	
Ineffective team building	Intermediate			
Acquisition management failures	Intermediate		Intermediate	
Ineffective government/industry teaming	Intermediate			
Real time efficiency decisions that led to later problems	Intermediate	2	High Frequency	
Dysfunctional feedback across the system lifecycle	Intermediate		High Frequency	
Using process rather than thinking and being accountable	Intermediate	1	Intermediate	

It is difficult to accurately compare the limited information we received on five failures at Company A to the extensive knowledge and expertise systems engineers have on failures at

Company A. However, the information in Table 2.13 does not contradict what the respondents indicated on the paper survey; we coded all of the findings in the failure narratives to issues the respondents identified as being high- or intermediate severity/frequency. However, some of the issues the respondents indicated were high-frequency were issues we did not identify in the failure narratives. This is likely due to us not having many failures to analyze and not having much depth into each failure.

In general, it is difficult to make conclusions on the Company A data because (1) we do not know how many systems engineers participated in the single paper survey response we received, and (2) we were only given limited data on five failures at the company. This may explain the discrepancies between the Company A data analysis, our analysis of past failures, and the results of the paper survey. For example, both our study of past failures and the respondents' paper survey results identified problems with risk management, but we only identified this cause in a single failure described by Company A (note that when we used the paper survey coding scheme, we identified "ineffective risk management procedures" in 3 failure narratives; our coding scheme allowed for more specific codes to be assigned to each finding). Overall, it makes sense that Company A provided supplementary data on failures they have experienced because many of the ideas represented in that data were difficult to translate to the paper survey.

2.6.3 Semi-Structured Interview Development

We wrote this semi-structured interview after Company B indicated they preferred interacting in a more free-form, organic environment. The general design of the interview is to ask questions at first meant to put the respondent at ease (e.g. about the respondent's educational background), then ask details about their experience as a systems engineer at the company (e.g. "what tools do you use?" and "what changes have you seen in systems engineering practices over time at your company?") and finish the survey by asking more difficult questions about problems they have encountered ("could you tell me about a shortfall you encountered that you found particularly difficult or interesting?"). We intended this format to make it easier for the respondents to open up about problems that contribute to project failures. Appendix A.2 contains the semi-structured interview questions in the order we asked them.

The semi-structured interview allowed us to ask questions about problems in systems engineering in general that contributed to other aspects of our research (e.g. areas systems engineers struggle with when they are first hired—see Section 3 for further discussion on this), as well as problems in systems engineering that lead to project failures (e.g. how they do or do not capture lessons learned). While this format does not allow us to directly compare its results to our study of past systems engineering failures, it did give us results that we describe next in Section 2.6.4.

2.6.4 Company B Responses

As described in Section 2.6.3, we collected responses to many different types of questions. In this section we discuss specifically the responses to questions related to (1) problems the systems engineers experienced on their projects and (2) how systems engineering efforts impact project performance. For further discussion on gaps in newly hired systems engineers' skillsets and weaknesses in systems engineering education refer to Section 3.1 and for further discussion on the tools the systems engineers used refer to Section 4.1.

We visited Company B and conducted the semi-structured interviews in person, interviewing two systems engineers at a time. We did 5 two-person interviews, yielding 10 responses. The responses we present in this section are not direct quotations, but rather have been paraphrased for ease of reporting and to obscure company-specific information. The questions from the semi-structured interview that relate to problems the systems engineers encountered and how systems engineering impacts project performance are:

1. What process changes have been implemented based on past projects?
2. How do you accommodate changes to a design after the process has begun? How do you make sure you don't have to change designs in the future?
3. Could you tell me about a shortfall that you encountered that you found particularly difficult or interesting and what factors contributed to this shortfall?

For each question we will compare some of the responses. For the full set of responses to each question, refer to Section B.1. Table 2.14 contains the responses to the question "What process changes have been implemented based on past projects?" We intended this question to identify what changes the company has implemented based on issues on past projects to improve project performance.

Table 2.14: Process changes that have been implemented on past projects

Respondent	Response
1A	Risk reviews are done differently, and now they elevate the reviews to the program manager in formal meetings. This company now makes risk reviews part of the staff meeting.
1B	There is far more emphasis on risk management over the last six months and an accurate perception that engineers are too optimistic in their scheduling. All the scheduling is optimistic and the company does not deliver products on time because engineers do not build time into the schedule. In general, there is no schedule float and all tasks are optimistically scheduled. The respondent is now working on a project that has a realistic schedule and a "best case" schedule. Engineers on that project design to the realistic schedule and hope for the "best case" schedule.
2A	Several years ago there was a directive to do documentation in DOORS. That required a lot of momentum to shift from Word to DOORS (because they had long-lead items, and did not know how to use DOORS).
3A	The company had varying processes that it changed to make more aligned; engineers now have to get internal approval before asking any of their customers whether to change a requirement. The goal is to get aligned internally before talking to customers.
3B	The company's compliance form process has become more formal. In the past their compliance tracking process was more ad hoc, but now the company has a document trail that accompanies submitted documents as evidence that requirements are met. Specifically, the form has specific areas where compliance is shown.
4A	20 years ago for a complex program the company had a requirements management tool they developed that mapped all the requirements to an analysis and had a complete checklist that showed that the design met every specification.
4B	The change management system has been improved and fairly well-standardized.

Many of the responses in Table 2.14 discuss specific, systems engineer-driven solutions to problems we identified in our study of past systems engineering failures. Respondents 1A and 1B discuss risk reviews and risk management processes that have changed based on problems with risks they encountered in past projects, such as schedule delays, which was a significant problem in the failures we studied. Respondents 2A, 3B, 4A, and 4B discuss how various documentation methods have become more detailed and how this helps the engineers keep track of aspects like requirements. We also found problems with requirements engineering and record-keeping in our study of past failures. Lastly, respondent 2A discussed how engineers need to get aligned internally before coming to one of their customers with a change. This solution would help problems with communication we found in our study of past failures.

Table 2.15 contains the responses to the question "How do you accommodate changes to a design after the process has begun? How do you make sure you don't have to change designs in the future?"

We intended this question to capture how systems engineers accommodate problems on projects after they have made progress in the design lifecycle.

Table 2.15: Accommodating a design change

Respondent	Response
1A	Lessons learned: the international programs are less structured than the US government programs. The US programs have a customer "breathing down your neck". International program customers are more "hands off". Now the company has a change review board so it can understand why there is a problem on a project and make sure they do the analysis properly so the problem does not happen again.
1B	Hopefully any mistakes made on a project can be rolled out as a lessons learned. It could be a useful tool in future discussions because it would show the consequences in cost and schedule for not doing things a particular way. It would also give systems engineering better standing to get management's attention.
2A	If a design does not meet certain requirements, it could indicate that a part of it is over-specified. If there is margin in the design, then change the requirement and make the paperwork match the design. It is a different story if the design actually needs to be changed.
4A	Pull the affected parties together, maybe one group missed a requirement or an interface interaction and they have to redesign. Then the groups discuss solutions (i.e. conduct trade studies) to get a best answer. Then get people to analyze it (people most appropriate for that specialty), and get internal and external permission. Formal change process makes the change official by getting change permission from the customer, in contractual or acceptance form. When the requirements specify making [a] change, some engineers believe the design's legacy is not going to change, but then there is no compatibility because one project changed and the other did not. The ones that did not implement the change did not realize the actual full impact of the non-change. Once the subsystem gets to a certain point, one group owns it and no one analyzes what could go wrong when all the subsystems interact. No one reviews the system as a whole and any little problem becomes problematic.
5A	When project managers identify that there is an issue, they come to systems engineers and ask for help to figure it out. They then sit down as a team and do an integrating event, involving all the major players to determine what needs to be done. There are also budgeting issues and the team has to figure out where the money comes from to fix it. It is exciting because the systems engineers get to do with "real" engineering work, in the nitty gritty of the technical aspects to redesign the part or whatever is causing the problem.

The responses for Table 2.15 contained some perspectives that were not reflected in our study on past systems engineering failures.

Respondents 1A and 1B were concerned with capturing lessons learned on a project. These respondents highlighted that lessons learned could be used to justify systems engineering efforts to show managers what the consequences of not doing systems engineering properly may be. One of the major causes in the systems engineering failures that we studied was that companies sometimes do not use the lessons they learned on failures they have encountered, so finding another use for the data they have collected could be a way to engage with this data the company has collected.

Respondent 2A said that components or subsystems can be “over-designed” and that sometimes the requirements need to be changed instead of the designs. We encountered this idea in our study of past failures, but it was not approached in the same way. Specifically, in the Future Imagery Architecture satellite project cancellation, Taubman [2007] criticized that the technology was not sufficiently advanced to accommodate what the engineers promised the satellite could do.

Respondents 4A and 5A each considered different perspectives when talking about how to solve this hypothetical problem of having to accommodate a design change. Respondent 4A pointed out that if one side of a project implements a change to a design once a problem is found, other sides may not implement those changes and this causes more trouble further into the design lifecycle when parts are expected to interface.

Table 2.16 contains the responses to the question “Could you tell me about a shortfall that you encountered that you found particularly difficult or interesting and what factors contributed to this shortfall?” We intended this question to directly compare some of the causes we found in our study of past systems engineering failures.

Table 2.16: Shortfall the respondents encountered

Respondent	Response
1A	A requirement on a project said that if personnel did an inspection, there was a maximum amount of time required to disassemble the system enough to gain access and perform the inspection. The respondent raised the questions: “How often do you need to inspect that? Is there a periodic inspection?” There was no means to inspect the area or give maintenance access to the area. Parts were on order for manufacturing and the design was finalized, so this problem required a significant rework. The engineers needed to redesign the area completely to accommodate an inspection hatch. The project was at risk of cancellation because of this issue. Contributors: Miscommunication and arrogance

Table 2.16 continued

1B	The company had an agreement with a customer to integrate an engine on to an aircraft. The engineers did not understand the implications of doing that because they did not find out what the requirements affected were. They should have figured that out before they started. The company had a customer that had a high technical content aircraft. The aircraft was complete and was in testing. The customer disagreed with the method the company had used to prove design success, and introduced requirements for additional tests. There was a big disconnect between the customer's requirement verification expectations and the company's. The company had to create a detailed verification matrix over the course of many months before the customer accepted the aircraft. The engineers did not start the job by asking how they were going to verify it. Contributor: Management overconfidence
4	Engineers on a project assumed a new system used the same data format as the previous system and although the new system was able to interact with the other systems, the interaction resulted in incorrect actions. They found that they had to analyze the interaction with the new system differently than the old one. Contributors: The engineers did not document lessons learned very well but they could have avoided repeating that scenario because there is tribal knowledge on that now.
5B	On a project, the location of a switch was not in a correct place on a subsystem. Human factors requirements were violated to keep it there, even though the customer wanted to keep it there and pushed for it to be located there. The more requirements you do earlier (even during a proposal) becomes hugely important. Get more upfront requirements work done as early as possible.

Recall that in Section 2.6.4 we noted that the data Company A provided did not discuss "people problems", such as problems relating to managing/supervising, training, communication, human factors, or team members' experience. In Table 2.16 we only showed the Company B responses that identified these problems to provide a contrast to the data we received from Company A. The respondents from Company B identify problems like miscommunication, overconfidence, arrogance, lack of imagination, and not documenting lessons learned, which skew more toward being labeled as "people problems". However, the respondents identify these people problems as contributing factors, rather than the primary cause of each problem (i.e. they do not say "miscommunication and arrogance led to a requirement being missed on a project"). Perhaps this difference exists because the respondents were speaking in a casual environment, rather than filling out a form. It is not surprising that different methods of data collection led to different types of responses and further application of both these survey methods may confirm these results.

2.7 Summary

Why do systems fail? In this section we wanted to determine whether we could link accident causation, which has been thoroughly studied over many decades and of which the systems engineering community has a sophisticated understanding, to project failure causation, which is less well understood. We began by reviewing accident research to understand the leading theories on what causes accidents. Then, to compare accident and project failure causation, we identified a set of 30 accidents and 33 project failures, spanning a wide range of industries. Next, we modified Leveson's STAMP model and used it to methodically extract and analyze the failure causes. We identified 23 different causes from these cases, which showed that accidents and project failures do indeed share causes.

We now have a more sophisticated way of understanding project failure causation by linking it to accident causation, but in what useful way can we leverage accident literature? We linked the causes we identified to recommendations that accident investigators made in the accident reports we studied, and identified 16 different recommendation remedial actions. We presented our findings as a cause-recommendation network, which shows how the failure causes are linked to each other and to the recommendations, and also provides over 900 specific examples of how these causes manifested in failures, and over 600 specific examples of the associated recommended remedial actions.

Are our findings similar to what systems engineers encounter on the problems they experience on projects? We interviewed experienced systems engineers at large-scale aerospace companies to determine how the problems they encountered compared to the problems we identified that led to systems engineering failures. Our two data collection methods yielded different results; one company identified fewer "people problems" such as management issues and problems with human factors. The other company allowed us to interview their systems engineers, and their responses added depth to our understanding of failure causation.

CHAPTER 3. ARE WE TRAINING STUDENTS IN SYSTEMS ENGINEERING WELL?

3.1 Results of Semi-structured Interview

In Section 2.6 we discussed the results of surveys we conducted with large-scale aerospace companies. In addition to gaining insights into failure causation in systems engineering, we collected some of the Company B systems engineers' criticisms on education. To see all of the responses we received for each of these questions, refer to Section B.2.

The questions from the semi-structured interview that relate to criticisms on systems engineering education are:

1. What is your general academic background and what schooling or other training did you receive before becoming a Systems Engineer?
2. Have you noticed any areas systems engineers struggle with the most when they are first hired?
3. Do you look for certain traits when hiring systems engineers?
4. Have you noticed any changes in systems engineering practices over time at your company?

Table 3.1 contains the responses to the questions "What is your general academic background and what schooling or other training did you receive before becoming a Systems Engineer?" We wanted to compare the background and education the systems engineers received with what they value in systems engineering education.

Table 3.1: Academic background and systems engineering training

Respondent	Response
2A	This respondent has an electrical engineering BS and MS. He did not undergo additional training to be a systems engineer, but has taken DOORS classes [a program used for requirements management].
3A	This respondent has a BS in Aerospace, MS in Aerospace, and an MS in system design and management. He has no formal systems engineering training. As far as the curriculum for his Bachelor's, there was no specific material relating to systems engineering. In his MS there was one course. Most of his specific systems engineering knowledge comes from being a pilot since he also did maintenance. He did some company-held courses, but mostly he has done a lot of on the job training.
3B	This respondent has a BS in mechanical engineering and an MS in Management of technology. For systems engineering, he has received on-the-job training.

Table 3.1 continued

4A	This respondent has a BS in Aerospace and an MS in systems engineering. He says his degree was technically called an Industrial engineering degree but this was before the university was allowed to call it "systems engineering." He also has an MBA. He was a Navy pilot (this included ground school and flying), so he has experienced use of the end product and knowledge of the systems.
4B	For systems engineering, he works just based on his experience, but he has attended occasional workshop-type courses and training on how to use systems engineering tools
5A	This respondent has a BS in Aerospace Engineering. For systems engineering, she's done mostly on-the-job training, with a majority of the training she received coming from the group responsible for system engineering processes.
5B	This respondent has a mechanical engineering BS degree with a Master's in management and a focus in product development. As far as systems engineering, he has learned a lot on-the-job, but there was training run by their systems engineering process group, which is responsible for organizational processes and training

We theorized in the introduction to this dissertation that the systems engineering community believes that successful systems engineers can only be made through experience, and that that may be due in part to experienced systems engineers not having much formal systems engineering-specific training in their undergraduate education. The responses in Table 3.1 show that most of the systems engineers we interviewed had gotten degrees in other engineering fields and then gained systems engineering experience on the job. We only included responses from respondents in this table who specifically mention how they received their systems engineering expertise, whether it be through on-the-job experience, schooling, or company-held courses. Respondent 4A got a Master's degree in "systems engineering" before it was called by that name. Five of the respondents spoke about receiving supplemental education by attending company-sponsored workshops that taught them about systems engineering tools and other topics related to systems engineering. Thus, their opinions on the usefulness of formal systems engineering education may be affected by attending these classes.

Table 3.2 contains the responses to the question "Have you noticed any areas systems engineers struggle with the most when they are first hired?" The responses to this question may highlight deficiencies in systems engineering education and provide contrast to what experienced systems engineers value versus what systems engineering educators value.

Table 3.2: Areas systems engineers struggle with

Respondent	Response
1A	There is a fine line between defining aspects at the aircraft level and the subsystem level (this is also a problem programmatically). It is hard to define the line, and also hard to define what the systems engineering scope is. Systems engineers' roles bleed in several directions.
1B	This respondent has two new employees right now. They struggle with clarity of communication; trying to communicate an idea unambiguously (like a risk or requirement). They need to define what they talk about in a way that cannot be misinterpreted and can be acted upon. He does not think they've been trained in clear communication because school teaches students to write things deliberately so their statements encompass a lot of ideas. That way the students can be correct no matter what when talking to a professor. It is a different environment in industry because clarity of communication is required to define the precision of the work. Another problem is that people usually give ambiguous work (i.e. work that is not readily assignable to a clearer area) to systems engineers because what the systems engineers do in the company is not clear to others.
2A	New hires have trouble with understanding the big picture. There is a stovepipe mentality that comes from new hires (fresh out of school): "I'm a mechanical engineer, I know this."
2B	New hires lack knowledge and experience across different disciplines of design. They are not familiar with what the product is. For instance, a new hire may have a mechanical engineering background, but being a systems engineer requires them to know about what electrical engineers need for their requirements. They may not know how to do the design work but they need to know what the designers consider, like environment, interfaces, signals, and structural loads (whatever the hardware or software is exposed to). Systems engineers have to know to ask the right questions.
3A	New hires do not understand this company's product. An understanding of aircraft, including their intricacies, coordination, and interfaces is needed. New hires' academic background is lacking in this regard, especially because this company has unique implementations and each program does everything a little differently.
4A	This respondent thinks new systems engineers have a lack of experience in the system as a whole. A single project at this company has so many different subsystems and subsequently how they tie together overwhelms new hires. Since this respondent was a Navy pilot, he already had first-hand experience with the product and this helped him have a much smoother transition than others. The systems produced by this company are complex and the new hires do not have the product experience. Understanding the system is difficult and even if they have the methodology they do not understand the interrelationship and functionality of the parts
4B	New hires do not understand what they're looking at when they see a requirement. The company has a lot of inexperienced people in general writing requirements. They can over-write (include too much detail) or under-write (include not enough detail). On the other hand, there are a lot of poorly designed requirements from the customer that newly hired systems engineers do not recognize are poorly designed. The requirements from the customer can be a statement of work and trying to translate that to performance requirement means it is untestable. In general, new hires don't get exposed to requirement writing. The writing discipline is not emphasized in engineering programs, nor is the interpretation of the requirements.

Table 3.2 continued

5A	New hires struggle with people skills and the ability to "own everything and not own everything." Systems engineers have a lot of disciplines they are responsible for but no one reports to them. They have responsibility but no authority. Being able to work as part of a large team is important for people who are in school to understand. Each person in a team has different skills, different functions, and different responsibilities. When this respondent was first hired, her platform was unfamiliar with systems engineers and saw them as in the way. People need to see the value of systems engineering.

Respondents 1A and 1B both spoke about problems inherent in systems engineering that lead to complications for new hires: that systems engineering is ambiguous and encompasses many roles, and that other engineers do not understand what systems engineers do and thus give them more ambiguous work that cannot be more specifically assigned to people in other engineering fields. Respondents 5A and 5B also spoke about how other engineers do not see the value of systems engineering or understand what they do.

Respondent 1B spoke about the education environment not fostering precise communication because students try to encompass many correct answers in their responses. Perhaps instructors should engage in more negative grading to discourage such shotgun approaches!

Respondents 2A and 2B both talked about new hires not seeing the "big picture" when it comes to their work. Notably, seeing the big picture is a trait identified by Frank [2012] as one of sixteen cognitive competencies successful systems engineers demonstrate. Respondent 1B stated that he believed this competency can only be learned through experience on the job. Respondent 2B thought that this problem was due to how engineering disciplines are parsed; a systems engineer charged with integrating a system may need to understand mechanical as well as electrical systems. It is currently difficult for a student to acquire skills under both disciplines without getting more than one engineering degree. Specifically, a student who graduated with a mechanical engineering degree may be weak in electrical systems and vice-versa. We agree that education for systems engineers should be more inter-disciplinary. Requiring engineering programs to collaborate more may also benefit systems engineering education by making it more consistent: a systems engineer graduating with a primarily electrical engineering degree should have the same skills and knowledge regarding systems engineering as a systems engineer graduating with a primarily mechanical engineering degree.

Respondent 4B noted in particular that newly hired systems engineers not only have difficultly writing requirements well but also recognizing good requirements. This finding gives more nuance to problems with requirements we identified in past systems engineering failures.

Respondent 5A is the only systems engineer to note a lack of "people skills" in new hires. Systems engineers typically work in teams and require skills to interface with other engineers, especially when those other engineers do not value systems engineering efforts.

Finally, many respondents said that new hires are not familiar with the company's products (e.g. 3A and 4A), which is certainly an aspect that formal education would capture with great difficulty (and we argue, likely should not—the goal of a university is not to equip a student for one particular job posting).

Table 3.3 contains the responses to the question "Do you look for certain traits when hiring systems engineers?" We wanted to identify whether traits experienced systems engineers look for when hiring systems engineers lean more towards innate traits or learned behaviors that education may be able to cultivate.

Table 3.3: Traits in new hires in systems engineering

Respondent	Response
1A	This respondent looks for whether a candidate can demonstrate the ability to focus and use the top-down approach. He wants to know whether the candidate has the ability to see the path ahead, and to execute the program in an efficient way.
1B	New hires should demonstrate flexibility and adaptability. The role of systems engineer is so varied and so broad; new hires have to be able to accommodate a variety of expectations and demands.
3A	This respondent looks for new hires to have a basic understanding of this company's processes and have a good academic foundation. He also looks for people being able to work in and interact with a team.
3B	New hires should have an open mind because some people focus on a single components. They should look at the system from an overall perspective and know how aspects interact with each other at a high level. Even if new hires don't have the ability to do that yet, they should have an openness to learn that.

Table 3.3 continued

4A	This respondent looks for people who are aggressive, assertive, and confident. Systems engineers have to work with other people, organize them, recognize potential problems and get people to work together. These people bring separate teams together to resolve situations and know how to do that.
4B	New hires should have creativity and independence.
5A	This respondent wants new hires who drive tasks to closure, especially because systems engineers have to ensure deliverables to customers within specific time frames. Adaptability is another good trait. Systems engineering involves an ever-changing workload and pressure from management. New hires need to find time to complete tasks within tight deadlines Dealing with various personalities of people on the project is another aspect of systems engineering.
5B	This respondent looks for a strong technical background, preferably with experience in requirements. He also looks for a sense of ownership and driving tasks to closure with accuracy and timeliness. Working well with a team is another trait he looks for because systems engineers have to deal with different personalities and take on a leadership role to pull together thing like design reviews.

Many of the ideas identified by the respondents are reflected in the literature on the attributes of successful systems engineers. Frank [2012] identified sixteen cognitive competencies that successful systems engineers demonstrate, and Derro and Williams [2009] identified behavioral competencies of highly regarded systems engineers at NASA, and aspects of both are echoed in the responses in Table 3.3. These aspects include understanding the whole system and seeing the big picture (Respondents 1A, 1B, and 3B), thinking creatively (Respondent 4B), possessing self-confidence (Respondent 4A), and building team cohesion and understanding the human dynamics of a team (Respondents 3A, 4A, 5A, and 5B).

Table 3.4 contains the responses to the question “Have you noticed any changes in systems engineering practices over time at your company?” We thought that responses to this question may give insight into how aspects of systems engineering education are outdated because of the ways in which the industry has changed over time.

Table 3.4: Changes to systems engineering practices over time

Respondent	Response
1B	The change in systems engineering practices is that the company now has them. These practices did not exist when this respondent started working at this company 30 years ago. In the beginning systems engineering did not exist. Expectations were simpler in the beginning. Often, customers did not explicitly define what they wanted. There was a lot of back and forth and it was a messy process that cost both parties money.
2A	There is now more consistency in systems engineering for big-picture ideas. Most of the DOORS [requirements management software] admins are consistent but some people are not admin but are doing an admin-type job. For example, a new program struggled with the DOORS infrastructure because they did not appoint an experienced DOORS admin. There is a trend at this company to use more DOORS, so more people know how to use it. At first this respondent was responsible for training a few people, but now he trains people at the whole company in how to use DOORS. In general, the technologies available are changing and people are becoming more familiar with it.
2B	People can spend days reading process procedures, and they end up not using them. People know where to go when they need details, but they don’t read the full report. When people go from one program to the next, for a given discipline there is consistency, but there is none for systems engineering.
3B	Upper management awareness has changed in that now they realize that systems engineering is necessary for projects to show compliance to their customer. This has led to more acceptance of how the systems engineers handle requirements and an improved “paper trail” for changing requirements.

Many respondents indicated that when they began at the company there was not a significant systems engineering effort and that it developed after the respondents were hired. In general, many of the respondents spoke about how systems engineering is becoming more accepted at various levels in the company, which makes it easier for them to interface with projects and enforce practices like good requirements writing.

Respondents 2A and 2B touched upon the consistency of systems engineering at their company Respondent 2B noted that other engineering disciplines have consistency, but there is none for systems engineering, which reflects its inconsistency across education efforts.

The responses to these questions confirm what we identified in the literature on systems engineering education: that experienced systems engineers received little formal training in systems engineering and believe that successful systems engineers can only be cultivated through on-the-job experience. The systems engineers we interviewed also described traits of successful systems engineers that are identified in the literature. Lastly, the respondents spoke about what has changed for systems engineering in the industry and highlighted that it is being taken more seriously as a discipline, which means that systems engineering education efforts may be more valued as time goes on.

3.2 Survey Development[6]

We have identified problems that lead to failures in systems engineering, but how can we prevent these from occurring in the future? One way is to determine whether our current systems engineering education efforts help students identify these problems by basing questions on failures we studied.

In Section 2.2, we compiled and classified the causal findings of 63 project failures across a variety of industries. We coded each finding as an actor-cause-object structure (e.g., “development management – conducted poor requirements engineering – requirements (safety)”). The “causal actions”, or “causes” are of particular interest in the context of testing skills, since they refer to the errors engineers made, and, hence, allude to missing skills or abilities. Table 3.5 shows relevant examples of findings and the resulting causal actions that highlight problems in systems engineering.

---

[6] We presented an earlier version of this work in Aloisio et al. [2018].

Table 3.5: Findings from our study of systems engineering failures

Finding(s)	Causal Action	Discussion/Explanation
Pike River Mine explosion: "The original mine plan specified two main fans located on the mountainside next to a ventilation shaft. Two planning changes were made. Pike decided to relocate the fans underground in stone at the bottom of a ventilation shaft. […] The decision was neither adequately risk hazards assessed nor did it receive adequate board consideration. A ventilation consultant and some Pike staff voiced opposition, but the decision was not reviewed. Putting the fan underground was a major error." [Panckhurst, 2012, p. 19] **Fukushima nuclear meltdown:** "When the Fukushima Daiichi station was constructed, the emergency diesel generators and emergency batteries were installed on the floor inside the plant building to afford protection against earthquakes. Ventilation ducts in the compartments where this equipment was located were not waterproofed. Moving this emergency power equipment to higher ground, safety experts said, would not have increased its vulnerability to seismic shock, provided it was fixed to a platform designed to resist earthquakes." [Action& Hibbs, 2012, p. 17]	Poorly managed risk Conducted poor requirements engineering	The mine operator decided to change an aspect of the ventilation system design, did not assess the risks associated with this decision, and thus did not consider how a potential explosion could be disastrous for the ventilation system. Similarly, the nuclear reactor designers, while designing safety systems for a reactor susceptible to earthquakes and tsunami flooding, did not consider how these disasters might affect the safety systems themselves.
SOHO spacecraft mission interruption: "Multiple ground operations procedures were modified. Each change was considered separately, and there appears to have been little evaluation performed to determine whether any of the modifications had system reliability or contingency mode implications; or whether the use of this modified procedure set should have been accompanied with operational constraints." [ESA & NASA, 1998] **Piper Alpha oilrig fire:** "Some of these additions [to the rig] apparently interfered with the proper functioning of safety features: external reinforcements on module C, for example, prevented adequate functioning of the blast relief. […] The result was that safety features that may have been adequate in the beginning became inadequate for this new layout, with new couplings and higher risks of accident that may not have been realized (or sufficiently questioned) at the time when the additions were made." [Paté-Cornell, 1993] "Although the structure itself was reinforced in 1979, the deck surface was fixed and the result of unpreplanned additions was an extremely packed space. Not only additional components were stacked, thus creating new couplings, but also, the recordkeeping of these additions was inadequate." [Paté-Cornell, 1993]	Failed to consider design interactions	For these failures, designers did not assess design changes for harmful interactions to the existing system.

Table 3.5 continued

Aloha Airlines flight 243 aircraft crash: "Aloha Airlines airplanes were accumulating flight cycles at twice the rate for which the Boeing MPD [Maintenance Planning Document] was designed. Even with an adjustment for partial pressurization cycles on short flights, and thus partial loading of the fuselage, the accumulation of cycles on aloha Airlines airplanes remained high and continued to outpace the other B-737 airplanes in the world fleet and Boeing's assumptions in developing the MPD." [NTSB, 1989a, p. 51]	Failed to consider customer needs	The aircraft manufacturer did not consider how its maintenance intervals would affect each specific customer; in the case of Aloha Airlines, the operator used the aircraft for frequent, short trips between the Hawaiian Islands.

Next, we identified a subset of these causes that involve scenarios that do not have an obvious "correct" response, lead to questions that students can answer in a short period of time[7], and have sufficiently detailed supporting information to provide a firm basis for creating a narrative. Many of the causes we identified did not fit these criteria, such as "failed to train" and "conducted maintenance poorly". The first cause describes poor training, like in the Texas City Refinery accident, in which operators did not follow procedures because they were not adequately trained in the plant's policies and procedures [Baker et al., 2007, p. 120]. The second cause describes poor maintenance efforts, like in the Three Mile Island accident, in which investigators found that the plant had a history of poor maintenance activities without adequate corrective action [Kemeny 1979, p. 47]. Both of these causes have obvious but simplistic answers: simply train personnel better and perform maintenance better. An adequate answer to a question framed around either of these causes is much more complex and requires comprehension of ideas like company management and company culture beyond that of an undergraduate engineering student, as opposed to comprehension of how systems work. Seven of the causes we identified did meet the criteria we described, so we used these causes to develop our survey questions. Table 3.6 contains these causes and their descriptions.

[7] To help ensure that we obtained a useful number of complete responses, we aimed for an average completion time of 45 minutes.

Table 3.6: Causes we based the survey questions on

Cause	Definition	Back story example
Created inadequate procedures	Actor(s) in the organization developed a deficient procedure, for instance maintenance, manufacturing, or emergency procedures.	Alaska Airlines flight 261 crashed because the maintenance personnel consistently did not lubricate the jackscrew assembly in the horizontal stabilizer properly, so the threads wore down over time. Since there were no threads holding the horizontal stabilizer in place, the pilots were unable to maintain pitch and the aircraft nosedived into the ocean. Among other causes, the procedures for a malfunctioning flight control system gave pilots unclear guidance and led to them improvising troubleshooting measures that could worsen the issue [NTSB, 2000, p. 140].
Conducted poor requirements engineering	Actor(s) in the organization did not lay out the needs, attributes, capabilities, characteristics, or qualities of the system well.	The V-22 is a unique aircraft that uses tilt rotors to take-off vertically, like a helicopter, and travel horizontally through the air like a turboprop aircraft. It was developed to fill a need to replace aging Marine helicopter transports, but can travel faster and farther than any helicopter used previously by the Marines. The program is over budget and behind schedule, however. Among other problems, the contract for the program had vague requirements, including not specifying an engine service life [Gertler, 2009, p. 9].
Failed to consider design aspect	Actor(s) in the organization failed to consider an aspect in the system design. In many cases, this causal action describes a design flaw, such as a single-point failure, improper system interactions, or component compatibility.	An explosion in a fuel tank shortly after takeoff brought down TWA flight 800. The NTSB concluded that a combination of a nearly-empty fuel tank and delaying the flight in July and having to run the air conditioning for the aircraft while it waited on the taxiway caused an explosive atmosphere to form. Once in flight, a short in the electrical system that measured the fuel levels in the tank ignited the atmosphere. It was common practice for aircraft to be flown with nearly-empty fuel tanks, which investigators thought was an avoidable risk. Among other causes, the placement of heat-generating equipment (e.g. the air conditioning system) under a fuel tank unnecessarily increased the amount of time the airplane was operating with a flammable fuel/air mixture [NTSB, 1996, p. 308].
Failed to consider human factor	Actor(s) in the organization failed to consider a human factor in system development. This causal action describes, for example, failing to consider human factors in specifying procedures or physical design.	The in-flight entertainment system was improperly installed on Swissair Flight 111, and as a result wires from the system chafed against metal components in the attic area of the aircraft. A spark started a fire on the aircraft while it was flying, and because it propagated in unoccupied parts of the aircraft, it went unnoticed and eventually brought the plane down. Among other causes, the standby instruments pilots used in an emergency were of a size and location that made them difficult for pilots to use, especially in a smoke-filled environment [TSBC, 1998, p. 254].
Failed to form a contingency plan	Actor(s) in the organization failed to form a contingency plan to implement if an unplanned event occurred.	The Exxon Valdez oil ship ran aground on the Prince William Sound in Alaska and because of the rocky bottom of the sound, many of the ship's cargo tanks were torn open and caused millions of gallons of crude oil to spill into the ocean. Among other causes, most of the emergency plans for an oil spill did not assume a spill of the magnitude of the Exxon Valdez spill. The plan that did prepare for the magnitude of the spill did not provide sufficient detail to guide the response [Skinner & Reilly, 1989, p. 8].

Table 3.6 continued

Managed risk poorly	Actor(s) in the organization failed to identify, assess, formulate, or implement a proper mitigation measure.	The Imperial Sugar refinery converted raw cane sugar into granulated sugar. The sugar was transported via a series of conveyors and elevators, which spread sugar dust throughout the plant. The sugar dust eventually ignited and caused an explosion. Among other causes, the facility's management were aware of sugar dust explosion hazards, but did not take action to minimize these hazards [CSB, 2008, p. 63].
Used inadequate justification	Actor(s) in the organization used inadequate justification for a decision.	Vioxx, made by Merck, is a non-steroidal anti-inflammatory drug that had widespread use to treat arthritis pain and inflammation. The drug was withdrawn from the market when a comprehensive study showed that people taking the drug had a significantly increased risk of heart attack. Among other problems, Merck attempted to explain away findings that Vioxx had a five times greater heart risk attack than a similar drug by claiming that the similar drug had an unproven protective effect, instead of acknowledging the risks and performing more studies or pulling the drug [Topol, 2004].

Using these back stories, we created a series of scenarios along with questions. We framed each question so as to obscure its origin while potentially allowing the student to draw out and discuss a decision error of systems engineering. Why not simply give students descriptions of the failures and the findings we discussed and have the students evaluate them? First, we wanted to eliminate bias due to students being familiar with a particular failure. For example, the Space Shuttle Challenger accident is a frequent topic in engineering ethics lectures. A learned, in-context, response from a previous exposure would not give us an indication of their abilities in systems engineering. Second, the point of framing a question around a decision error is not to, for example, discuss whether they would launch Space Shuttle Challenger, but instead discuss what else they would consider in the launch decision. The more open-ended question may give us more insight into the student’s thought process. Table 3.7 contains descriptions of the two survey questions we discuss results from in Section 3.4.3 and what aspect of systems engineering we expected each question to test. Refer to the Appendix for the same descriptions of the remaining 6 survey questions.

Table 3.7: Survey question descriptions

Survey Question/ Accident	**Description**	**What it tests**
Flood Wall Question New Orleans levee collapse [ASCE, 2007]	Cause: Used inadequate justification for project design, Conducted poor safety requirements engineering Decision error: designers did not consider the interaction of the sand substrate, water, and wall design that caused the wall to easily tip when the water saturated the substrate. Question format: the question prompts students with design principles such as “absorbs damage” and “contains functional redundancy” from Jackson & Ferris [2013] and asks the students which design principles the flood wall design satisfies.	This question presents students with a flawed design and gives them tools to criticize it as well as improve it. The student must determine a design principle the flood wall satisfies, and then improve the design by selecting a single best design principle to incorporate into the flood wall design.
Oilrig Question Piper Alpha oil spill [Cullen, 1990]	Cause: Failed to consider design interactions Decision error: the personnel quarters were not designed with access to emergency controls or equipment such as life boats, and subsequently personnel, who were instructed to wait there during an emergency, were trapped. Question format: As in the levee wall question, the question prompts the students with design principles and asks them which design principles the oilrig design satisfies.	As with the flood wall question, the oilrig question presents students with a flawed design and gives them tools to criticize it as well as improve it. The student must determine a design principle the flood wall satisfies, and then improve the design by selecting a single best design principle to incorporate into the flood wall design.
Boat Race Question Challenger Space Shuttle explosion [Rogers, 1986] [James, 2015][8]	Cause: Managed risk poorly, Used inadequate justification for quality issue Decision error: The crew decided to launch the Challenger Space Shuttle, despite evidence suggesting potentially catastrophic damage to the vehicle because a crucial component was vulnerable to below-freezing temperatures and doubt on the success of the launch from experts on the program Question format: The question describes an imaginary scenario about a boat racing team experiencing various failures all season and presents a decision point on whether to continue racing despite cold temperatures on the day of the race. The question asks the student what other factors the crew should consider when deciding whether or not to race—what could be contributing to the failures the crew is experiencing.	This question gives very little detail and this allows students to consider the system as widely as they wish (e.g. the engine, the boat as a whole, the driver and the boat, the humans interacting with the boat and the boat). The students are not simply rewarded for making a decision on whether to race, but rather on how deeply and broadly they thought about the system.

[8]This survey question was also partially inspired by “Carter Racing” described in James [2015], although the final format of this question is different than the inspiration.

Table 3.7 continued

Empennage Question Alaska 261 aircraft crash [NTSB, 2000]	Cause: Failed to consider human factor (equipment design), created inadequate procedures Decision error: the t-tail configuration of the aircraft's empennage required specific maintenance, which was difficult to perform because it was not easily accessed/visible, causing the component to wear out and fail during flight. Question format: The question provides the students with a diagram of an aircraft empennage configuration as well as a table comparing two common empennage configurations. The question asks the students to rank three categories: "maintenance", "performance", and "safety" in terms of what they think are the most important, then discuss the designs based on their ranking judgement and compare the designs in a short paragraph.	The question presents advantages and disadvantages for each empennage design and the student must weigh the trade-offs in deciding between the two designs and justify why they selected that design.
Aircraft Maintenance Question Aloha 243 aircraft crash [NTSB, 1989a]	Cause: Failed to consider design aspect (customer needs) Decision error: Aloha Airlines used their aircraft to travel between Hawaiian islands, which is a relatively short trip. The maintenance intervals for the aircraft were not tailored to the needs of the airline, and were based on flight hours instead of number of flights, causing the fuselage to fatigue faster than usual. Question format: The question describes an imaginary scenario in which an aircraft that was previously used on long flights is not being considered for short flights. The question provides the students with a description of five aircraft systems (landing gear, engine, fuselage/cabin, landing flaps and spoilers, and electronic systems) and asks the student to identify and discuss which systems may require different maintenance programs with this different application.	The students must discuss which systems are affected by the change and why. The students have to think about the difference the two types of routes and the implications of those differences for the aircraft and its subsystems.
Off-Road Vehicle Question Ford Explorer vehicle quality issues [Bradsher, 2000]	Cause: Used inadequate justification for quality issue Decision error: The car manufacturer's decision to make insufficient post-design modifications to the vehicle when they found the vehicle was unstable during testing. Question format: The question describes an imaginary scenario about an off-road vehicle that failed a stability test and presents the student with four solutions to this problem: (1) adding a large plate under the vehicle, (2) lowering the cabin by replacing the suspension system, (3) redesigning the entire vehicle, or (4) changing the tires to slightly increase stability. The student then must rank each solution in terms of safety, cost, marketability, and time to complete, rate each of these categories on a scale of relative importance, choose a solution, and discuss why they chose that solution.	Students must consider trade-offs in the design and clearly articulate their priorities. The student has to discuss their decision and ensure their discussion matches their trade-off choices. Systems engineers frequently use tools to rank systems and then make their decisions based on the ramifications of the outcomes of using those tools, not on their "gut instinct".

Table 3.2 continued

Toothbrush Requirement Question Requirements engineering problems noted throughout our study [Aloisio & Marais, 2017a]	Cause: Conducted poor requirements engineering Decision error: we identified problems with requirements engineering throughout our study of systems engineering failures. For example, in the Pike River coal mine collapse the requirements for ventilation system were not adequately defined; the main fan was placed underground and was not explosion-protected, and thus immediately failed during the initial methane gas explosion [Panckhurst, 2012]. Question format: The question provides students with four requirements for a toothbrush and asks them to specify two "terrible" features for the toothbrush (i.e. features that make the toothbrush unusable) that fit within these requirements. The students must then write a requirement that prevents at least one of the features from being incorporated into the toothbrush.	This question reverses the students' typical requirement-writing process by having them imagine the worst design and write requirements to prevent that, rather than having an ideal design in mind while writing requirements to supplement that. An important aspect of the question was prompting the student to write an adequate requirement.
Vehicle Maintenance Shop Question Piper Alpha fire [Cullen, 1990] SOHO communication loss [ESA & NASA, 1998]	Cause: Conducted poor requirements engineering, Failed to consider design interactions, Failed to form a contingency plan Decision error: The Piper Alpha oilrig design was significantly modified decades after it was put in service to incorporate additional equipment, living quarters, and crew amenities. These design modifications interfered with the functions of some safety features included in the original design, and there were unforeseen design couplings. Question format: The question provides students with a scenario in which a maintenance shop is considering providing transmission repair services. The students are asked to consider the impact of making this change on employee training cost/time, shop resources cost/time, eliminating the "middle man" cost/time, and marketability profits. The students must rank each of these aspects in terms of importance to the shop's success and discuss whether offering transmission repair is worth the modification	This question challenges the student to consider what may occur to an existing system when significant structural changes are made.

3.3 Survey Deployment

To date, we have distributed our survey in four semesters (Fall 2016, Spring 2017, Fall 2017, and Spring 2018), and analyzed the responses from two of those distributions (we are now grading the Fall 2017 responses and waiting to receive the results of the Spring 2018 responses). We distributed the survey online using the Qualtrics survey platform through email to Purdue students in Aeronautics and Astronautics (AAE). For each distribution the survey was available for two weeks. We incentivized the students to participate in the survey each semester by offering them an opportunity to enter a random drawing for a $100 Amazon gift card.

Each time we deployed the survey, we sent a copy of the survey through Qualtrics to a representative at Purdue's Center for Instructional Excellence, who emailed the students in AAE and asked them to take the survey. The representative sent this email multiple times over a two week period, then closed the survey and connected each response to the student's relevant Bursar data, anonymized the data, and returned it to us. We used this method so that we would not know which students had participated in the survey to avoid complications (the principle investigator on this project also teaches courses in AAE and may unconsciously be biased toward favoring students who participated in the survey) and so that we would not have unnecessary access to data we do not need, such as which elective courses a student has taken.

Along with the survey responses, we also collected responses to personality-type questions relating to systems engineering ability and academic performance and academic data (overall GPA, and what grade they received in specified systems engineering-related courses at Purdue). "Systems engineering related courses" are all courses designated as "design" or "systems" in the School of Aeronautics and Astronautics. These courses contain systems engineering-related tasks, such as writing requirements, designing, and design verifying. Courses such as senior-level design-build-test courses and the sophomore-level introduction to aeronautics and astronautics course are thus included in our data. We also collected student demographic data (gender, age, ethnicity, and student classification) because past research has indicated that male and female students may perform differently and we wanted to investigate how these differences manifested for our survey. We received a total of 275 responses to the survey, and Table 3.8 describes these responses. The Purdue Aeronautics and Astronautics department has approximately 1,000 students (~400 graduate,

~600 undergraduate) enrolled in each semester. That means that the response rate fluctuated between about 5% and 10% each semester.

Table 3.8: Breakdown of the survey responses

Group	Subgroup	Number of responses
Student classification	Graduate	133
	Undergraduate	142
Distribution Semester	Fall 2016	47
	Spring 2017	97
	Fall 2017	49
	Spring 2018	82
Completeness	Complete responses (all 8 survey questions)	157
	Incomplete responses (fewer than 8 survey questions)	118
Average GPA of students who responded to the survey	Graduate	3.57
	Undergraduate	3.32
Student sex as identified the Purdue registrar	Male (graduate)	114
	Female (graduate)	28
	Male (undergraduate)	113
	Female (undergraduate)	20
Average number of systems courses student has taken	Graduate	2.2
	Undergraduate	1.5
Median number of systems courses student has taken	Graduate	0
	Undergraduate	1

Table 3.8 shows that we received approximately the same number of responses from graduate and undergraduate students (which means that a higher percentage of the graduate population responded than the undergraduate population), and that we received more responses in the Spring Semester than the Fall Semester (which may have implications for how we distribute the survey in the future). We received many complete survey responses. The undergraduate students who responded to the survey had an average GPA of 3.32 [Crain, 2018a], which is approximately equal to the average undergraduate student GPA in Purdue AAE, and indicates that our sample may not be biased towards students who receive high grades. Additionally, Purdue AAE’s undergraduate student population is 13% female [Crain, 2018b], while its graduate student population is 17%

[Delaney, 2018]. Undergraduate students who responded to our survey were 20% female, while graduate students who responded to our survey were 15% female. Based on these comparisons to the Purdue AAE student population, we have no reason to believe that there is a difference between the general student population in Purdue AAE and the survey respondents.

Graduate students who responded to our survey took an average of 2.2 systems courses, but the median of the number of systems courses they have taken is 0, which indicates that there are a few graduate students who have taken many systems courses, but most have not. The undergraduate students who responded to our survey took an average of 1.5 systems courses, and the median number of systems courses they have taken is 1, which indicates that many of the undergraduates have taken just one systems course.

3.3.1 Grading Scheme

We initially created a grading rubric and graded each survey question as an “A”, “B”, or “C” response. Since we anticipated short essay responses, we used a limited-resolution grading scale that is widely used by faculty. However, we found that the A, B, C grading scheme had too much variability within grades and was difficult to administer objectively in many cases. We thus changed the grading scheme to a more objective, quality-based method of grading. Each question was graded out of 8 qualities, with 3 qualities derived from competencies of successful systems engineers that we applied to each question (although with slightly different specifications), and 5 qualities that were question-specific. We marked the student a “1” if their response met a certain quality, and a “0” if it did not. Table 3.9 shows the quality-based grading scheme for the boat race question. Appendix C contains the grading schemes for the other survey questions.

Table 3.9: Improved survey grading scheme for boat race question

#	Quality	Explanation
General response qualities		
1	Student's response contains more ideas than those given in the question description. Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].	Responses ONLY containing these ideas do NOT have this quality: Aspects related to temperature (weather, water/air temp) Problems with engine components, like fuel, oil, parts, engine systems, etc. Problems specifically related to sponsorship loss If the student only discusses these ideas, it is unclear whether the student is truly considering "other factors" the crew should consider.
2	Student's response reflects cognitive competencies that successful systems engineers possess. Justification: Systems engineers have certain cognitive competencies [Frank, 2012]. Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].	The student's response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question:
		Understand the whole system and see the big picture: "understand the whole system beyond its elements, sub-systems, assemblies and components, and recognize how each element functions as part of the entire system."
		Understand interconnections: "understand the interconnections and mutual influences and interrelations among system elements."
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems."
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level."
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system."
		Able to take into consideration non-engineering factors: the student considers non-engineering factors related to the question.
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	The grader understands what the student is communicating with minimal confusion or conjecture. This does not necessarily mean that the response is completely spelling- or grammar-error free.

Table 3.9 continued

Question-specific response qualities		
4	Student identified a relevant factor the crew should consider.	The student identified a factor that could contribute to the boat failure or that would help the crew consider whether or not to go through with the race. An “irrelevant” factor is a factor the crew should not necessarily consider when determining the source of engine failure or whether to follow through with the race, or a factor that is not specific enough to make a relevancy judgement on.
5	Student identified more than one relevant factor the crew should consider.	Listed more than one relevant factor, even if they are related to each other.
6	Student-identified relevant factors relate to more than one aspect of the system.	Factors do not all relate to engine components or atmospheric qualities like air temperature/water temperature.
7	Student describes how factors relate (or do not relate) to poor performance in the system.	The student demonstrates that they understand how this factor would be related to engine failure or would help the crew make the decision of whether or not to race.
8	Student proposes ideas for how to investigate or fix a factor they identify.	The student proposes either (1) how the crew should verify whether this factor is related to engine failure, or (2) how the factor could be addressed before the boat race.

3.3.2 Inter-Rater Correlation

How objective is our grading scheme? Inter-rater analysis is a method used to determine the objectiveness of rating schemes by demonstrating consistency among observational ratings provided by multiple coders [Hallgren, 2012]. To save time and resources we proposed having an additional grader use the grading scheme for a subset of responses for each survey question as opposed to having the grader rate each response. How many responses does the additional grader need to rate to provide sufficient statistical power to make conclusions about the objectiveness of the grading scheme? There is not much guidance in the inter-rater literature on grading subsets of responses, so we decided to measure the consistency of grades on a subset of responses, rated by two graders, using a correlation test. This required us to make some assumptions to determine a sufficient basis for the number of responses necessary to perform this test. Our assumptions are as follows:

1. We compare the overall scores (e.g. 6 out of 8) using a linear model. In our analysis of the survey data we use linear and ordinal regression to compare the results and the conclusions on the effectiveness of those models is discussed in Section 3.4.1.

2. We use a correlation test to determine whether the grades are correlated, and a higher correlation score indicates that the two graders scores are more similar or consistent[9], and thus the grading scheme is more objective.

Our null hypothesis is that the grading scheme is objective, and that the correlation between graders is high (between 0.85 and 0.95). Figure 3.1 shows a correlation value of 0.85 between the graders. For demonstration purposes, the values in this figure are simulated values, and are continuous (not integers, like our grading scheme yields). So, we interpret that any values that fall outside the blue dotted bounds on the lines indicate grades different enough for the graders to assign different integer values. As the correlation value increases, more of the grades fall within the bounds, and this indicates that the graders are assigning more similar grades.

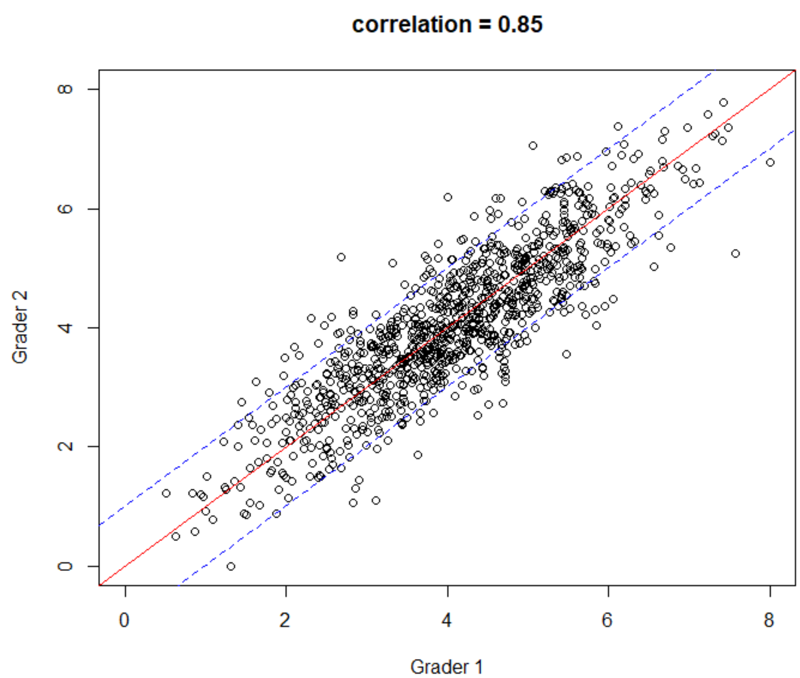


Figure 3.1: 0.85 Correlation between two graded sets of responses

[9] This means that the graders do not necessarily have to give the same grade, as long as they grade each response consistently. If Grader 1 assigns scores higher by 1 point than every score Grader 2 assigns, then the grading is still correlated. This is because the significance in the linear regression model is not affected by consistent differences between scores; the coefficients will have different magnitude by some scaled value (e.g. if the other grader scores twice as high, the coefficient will be twice as much). Further explanation of the results of linear regression model is in Section 3.4.1.

We can figure out the correlation of two sets of responses, but we still need to determine how many responses is sufficient to reject the null hypothesis; that is, to detect whether the grading rubric creates uncorrelated grades. To do this, we conducted a power analysis. Figure 3.2 shows a power analysis for different sample sizes and real correlations with a null hypothesis of 0.85 correlation. The power indicates how effective a correlation test is at detecting whether we can reject the null hypothesis. For example, if the real correlation of the two sets of grades was lower than we assumed, such as 0.7, we would not need many responses to determine that we need to reject the null hypothesis. If the real correlation is close to the assumed value, such as 0.8, it would take many more graded responses to determine the difference between the real and assumed correlation. The black horizontal line in this figure indicates 60% power, which means that there is a lower chance of capturing the difference between the null hypothesis and the real correlation. The number of responses we need the additional grader to rate at 60% power is 15 to capture differences up to 0.65.

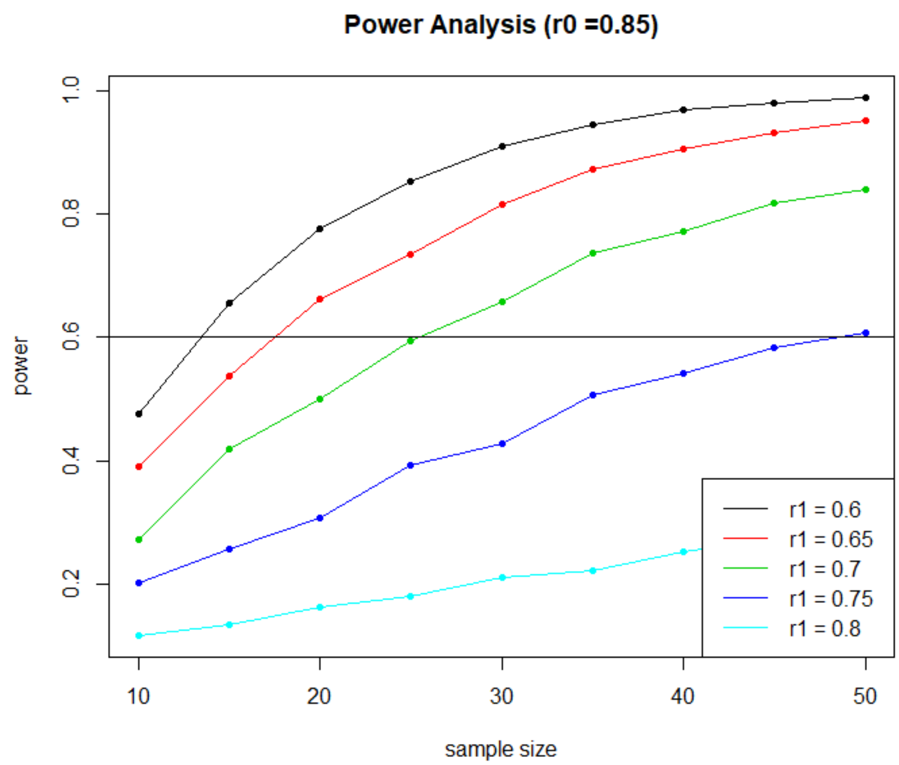


Figure 3.2: Power analysis for different sample sizes and correlations with real correlation = 0.85

Based on our power analysis, we had the additional grader rate 15 randomly-selected responses for each question and compared the responses to the ones we rated. Table 3.10 contains the results of the correlation analysis for each of the survey questions. Green shaded boxes indicate correlation values close to what we assumed the test would detect, and the red boxes indicate correlation values much lower than we anticipated.

Table 3.10: Correlation and p-values for each survey question

Survey Question	**Correlation**	**P-value**
Empennage	0.653	0.016
Transmission	0.735	0.109
Toothbrush	0.889	0.677
Car Design	0.556	0.008
Oilrig	0.844	0.430
Levee	0.640	0.018
Boat	0.864	0.547
Aircraft Maintenance	0.710	0.078

A null hypothesis correlation of 0.85 yields a minimum sample correlation to reject the null hypothesis of 0.71. This means that the values highlighted in green indicate correlations that are “high enough” and that we do not reject the null hypothesis for those questions, meaning that as far as our statistical test can detect, our grading rubrics for these questions are objective. For three questions (Empennage, Car Design, and Levee), we are able to reject the null hypothesis and this may indicate that the grading rubrics for these questions are no objective and may require more work to make them more objective. How exactly did the graders differ with these questions, and where may the grading scheme be improved? Table 3.11 contains the number of times each grader differed for each quality on the grading rubric for the car design question.

Table 3.11: Differences between scores for Car Design responses

#	**Quality**	**Number of times Grader 1 and Grader 2 differed**
1	Student’s response contains more ideas than those given in the question description.	5
2	Student’s response reflects cognitive competencies that successful systems engineers possess.	7
3	Student’s response is clearly communicated.	1

Table 3.11 continued

4	Student ranked the designs in each category and assigned point values to each category.	0
5	Student selected an option.	1
6	Student's ranking and selected option do not contradict each other (+/- one rank).	1
7	Student response contains discussion on why they chose that option by describing why the other 3 design options are worse for any aspect.	0
8	Student discussed each of the four categories given for the design option they selected.	2

The information in Table 3.11 indicates that some aspects of the grading rubric of this question require more attention than others. Table 3.12 contains the changes we made to the car design grading rubric for each quality.

Table 3.12: Changes we made to the grading rubric for the car design question

#	Quality	Original Grading Rubric	Improved Grading Rubric
1	Student's response contains more ideas than those given in the question description.	Example of a response with an "extra" factor: If this is a custom vehicle for a certain operator with specific requirements and at an early stage, it would be redesign from scratch. However, if this is for mass markets and just before the mass production, changing of tire pressure and written instruction and/or warning on manual might be the one.	The student's response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality. Example of a response with an "extra" factor: If this is a custom vehicle for a certain operator with specific requirements and at an early stage, it would be redesign from scratch. However, if this is for mass markets and just before the mass production, changing of tire pressure and written instruction and/or warning on manual might be the one. Other examples include the type of customer who would purchase an off-roading vehicle, what the fundamental differences are between off-roading vehicles and everyday vehicles, the market for vehicles, and what the consequences are for vehicles tipping.

Table 3.12 continued

2	Student's response reflects cognitive competencies that successful systems engineers possess.	The student's response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question.	The student's response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question. Typically the cookie-cutter, bland student responses do not display this quality. The student's response must demonstrate that they thought about the problem in a "systems engineering" way.
3	Student's response is clearly communicated.	The grader understands what the student is communicating with minimal confusion or conjecture. This does not necessarily mean that the response is completely spelling- or grammar-error free.	The grader understands what the student is communicating with minimal confusion or conjecture. This does not necessarily mean that the response is completely spelling- or grammar-error free. Did you understand right away what the student was trying to say?
4	Student ranked the designs in each category and assigned point values to each category.	The student assigned rankings to each design in each category (i.e. each category doesn't have the default ranking, with Option 1 being first and Option 4 being fourth) and assigned point values to each category.	The student assigned rankings to each design in each category and assigned point values to each category. If there are any blanks for the point value assignments, or all four categories have the default ranking (i.e. 1, 2, 3, 4), the student's response does NOT have this quality.
5	Student selected an option.	In their discussion, the student selected an option the design team should move forward with in the design.	In their discussion, the student selected an option the design team should move forward with in the design. Responses that discuss the "top two" designs are also acceptable.

Table 3.13 continued

6	Student's ranking and selected option do not contradict each other (+/- one rank).	The grader calculates an adjusted rank based on how the student ranked each design option and the point values the student assigned to each category. In their response, the student should have selected an option that has an adjusted rank of 1 or 2. If the student selected an option that the researcher calculated to have an adjusted rank of 4, for example, it would indicate that there is a contradiction between the rankings/point values the student assigned and the design option the student selected.	The grader calculated an "adjusted rank" based on how the student ranked each design option and the point values the student assigned to each category (columns AD through AG have a formula that calculates this and displays it as a rank). For ease of interpretation, the excel sheet has conditional formatting that shows the lowest-ranked design in red and the highest-ranked design in green. If the student's response has met quality 5 and they have selected the best design in their opinion, compare this selection to the "adjusted rank" they gave this (or these) designs. The design they selected should have an adjusted rank of 1 or 2. If the student selected an option that the has an adjusted rank of 4, for example, it would indicate that there is a contradiction between the rankings/point values the student assigned and the design option the student selected and their response does NOT have this quality.
7	Student response contains discussion on why they chose that option by describing why the other 3 design options are worse for any aspect.	The student discussed why they chose a particular design option over the other 3 (they must describe some aspect of all 4 designs in their discussion).	The student discussed why they chose a particular design option over the other 3, and they must describe some aspect of all 4 designs (by name) and in some form in their discussion.
8	Student discussed each of the four categories given for the design option they selected.	The student discussed each of the four categories (safety, monetary cost, marketability, and time to complete) for the design option they selected.	The student discussed each of the four categories (safety, monetary cost, marketability, and time to complete) for the design option they selected.

After rewriting the grading rubric to be clearer for the car design, empennage, and levee survey questions based on the differences in grades, we had another grader rate another subset of responses to determine whether the correlation improved. Table 3.13 contains the correlation and p-values for the improved grading rubrics. These values are now within acceptable limits and we cannot prove that our grading rubric is not objective.

Table 3.13: Improved grading rubric correlation and p-values

Survey Question	**Correlation**	**P-value**
Empennage	0.768	0.138
Car Design	0.783	0.172
Levee	0.840	0.4024

We calculated the correlation and conducted a power analysis to determine an appropriate minimum number of survey responses we could have the additional rater's grade, but the correlation we calculated does not indicate specifics of each grade. A high correlation value could indicate that one rater is grading responses 3 points higher, for example, as long as the two raters graded consistently, and our analysis so far has indicated that our grading scheme does not produce inconsistent grades between raters. How well do these ratings align when we directly compare them? Using a different analysis method, can we provide evidence that our grading scheme is not objective?

To determine the answers to these questions, we calculated the concordance between each grader. For each quality for every survey response, we compared the grades between raters. If these values matched, we assigned a "1" to this quality. If not, we assigned a "0" to this quality. We then summed the values and divided by the total number of qualities we compared. Table 3.14 shows the concordance value as a percentage for each survey question, as well as individual concordance values for each quality. Higher percentages indicate a higher agreement between raters.

Table 3.14: Concordance between raters for each survey question

Survey Question	**Concordance**								
	Overall	**General Qualities**			**Question-Specific Qualities**				
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Empennage[10]	70%	80%	80%	80%	75%	45%	85%	45%	70%
Transmission	73%	60%	47%	93%	80%	80%	80%	93%	47%
Toothbrush	79%	87%	53%	73%	93%	93%	93%	60%	80%
Car Design	86%	80%	45%	85%	95%	100%	95%	100%	85%
Oilrig	75%	55%	50%	100%	50%	100%	90%	75%	80%

[10] For the Empennage, Card Design, and Levee questions we used the grades from the second set of raters.

Table 3.14 continued

Levee	77%	90%	80%	70%	90%	60%	60%	80%	85%
Boat	85%	70%	70%	95%	95%	95%	80%	80%	95%
Aircraft Maintenance	77%	53%	7%	87%	93%	100%	100%	87%	87%

The overall concordance values are high, indicating that we cannot prove that our grading scheme is not objective using this analysis method. However, the concordance values for some of the specific qualities are low (as low as 45%), indicating where we should improve the grading scheme to make it more objective in the future.

3.4 Response Analysis

Does performance on the survey correlate with information we collected on how students are learning systems engineering concepts in Purdue Aeronautics and Astronautics? To determine the answer to this question, we performed multiple analyses on this data. We begin with linear and ordinal logistic regression, the first of which requires us to make assumptions that may or may not be valid. The second regression method does not require us to make the assumptions as for linear regression but is more difficult to interpret. Next, we use classification and regression trees to visualize the data. Finally, we analyze the data qualitatively.

3.4.1 Regression Analysis

We want to know whether the student's survey performance is affected by variables we measured that correspond to how student are learning systems engineering concepts. Table 3.15 contains these variables, their descriptions, as well as how we will represent them in formulae.

Table 3.15: Variables we considered in our analysis

Variable	**Representation**
Student's grade point average (GPA)	x_{GPA}
Student's self-identified confidence in systems engineering abilities such as requirements writing and systems verification	$x_{confidence}$
Student's average grade in courses we identified as having material related to systems engineering, such as design-build-test courses	$x_{GPA_{SE}}$
The number of courses the student has taken that we identified as having material related to systems engineering, such as design-build-test courses	x_{NSE}
Whether the student has been identified as male or female in Purdue's administrative system[11]	x_{MF}
The semester the student was considered in when they took the survey (i.e. freshman year, semester 1 is 1 and junior year, semester 1 is 5), as determined by their cumulative credit hours	$x_{classification}$[12]

[11] At this point in time the Purdue registrar identifies students as either male or female in their demographic data. Students' gender identities therefore may not be captured accurately by this field. We are more interested in the behavior of this variable in interaction terms rather than this variable on its own.

[12] This variable is considered in the classification and regression tree analysis in Section 3.4.2.

We analyzed the results of the survey using the linear regression model with the "lm" function in R. Linear regression is used to determine the linear relationship between the response variable and the explanatory variables [Prabhakaran, 2016]. However, to model our data in this simplified way we first need to assume that the levels between each of the scores is the same. That is, if a student received a score of 6 on a survey question, their response performed exactly twice as well as a student who received a score of 3 on the same survey question, regardless of which qualities each response demonstrated. Equation (2) describes the linear relationship between the outcome variable Y and the predictor variables contained in the array $\boldsymbol{x}$ with intercept α and $\boldsymbol{\beta}$, an array containing the slopes for each predictor variable, and error term ε.

$$Y = \alpha + \boldsymbol{\beta x} + \boldsymbol{\varepsilon} \tag{2}$$

We interpret Equation (2) as Y being linearly proportional to these variables ($\boldsymbol{x}$). If the slope (β) of a given variable is **positive**, that indicates Y will **increase**. For example, if β_{NSE} (the slope of the variable describing the number of systems engineering-related courses the student has taken) is **positive**, the model indicates that as the corresponding variable (x_{NSE}) increases, indicating that the student has taken more courses relating to systems engineering, the student will receive a **higher** survey grade.

To illustrate this model, Equation (3) contains the values for the linear regression on the undergraduate responses to the transmission question.

$$Y = 1.36 - 0.19\, x_{GPA} + 0.22x_{GPA_{SE}} + 0.06x_{NSE} + 1.01x_{confidence} + 0.39x_{MF} \tag{3}$$

Equation (3) expresses that the student will receive a higher grade on the boat race question as $x_{GPA_{SE}}$, $x_{confidence}$, and x_{NSE} increase, and as x_{GPA} decreases, and if they are a male student. The variable representing the students' self-confidence in their systems engineering abilities, indicating that this variable has the highest impact in the model.

We also analyzed the results of the survey using the proportional odds model for ordinal logistic regression with the "polr" function in R. Ordinal logistic regression is used when the data contains ordinal categories but it cannot be assumed that the difference between each level is considered the same. The proportional odds model is further detailed in McCullagh [2013] and described by equations (4) and (5).

$$\text{logit}\left(\gamma_j\right) = \log\left(\frac{\gamma_j}{1-\gamma_j}\right) = \alpha_j - \boldsymbol{\beta}^T \boldsymbol{x} \tag{4}$$

Where

$$\gamma_j = P(Y \leq j \mid \boldsymbol{x}) \tag{5}$$

How do we interpret this model as compared to the linear regression model? Applying the equations, in words (3) becomes: γ_j is the probability of receiving a survey grade (Y) less than value j (i.e. a grade of 4 or below out of 8), given the presence of variable x (i.e. x_{NSE} or $x_{GPA_{SE}}$). Equation (4) relates the proportional log-odds of variable γ_j to a linear equation with intercept α_j and slopes $\boldsymbol{\beta}$ of variables $\boldsymbol{x}$ (i.e. x_{NSE} or $x_{GPA_{SE}}$). Each slope (β) indicates what effect the corresponding variable has on the logit equation. **Positive** β values indicate that as x increases, logit(γ_j) decreases, meaning that the probability of getting a survey grade (Y) less than a 4 or below (j) decreases; thus, there is a **higher** probability of the student getting a better survey grade. The polr function compares each level between the grades the students received on a certain question, without needing to assume that the difference between each level is equal as the linear regression model does. Figure 3.3 describes how the function displays each level comparison and how that comparison translates to the grades[13].

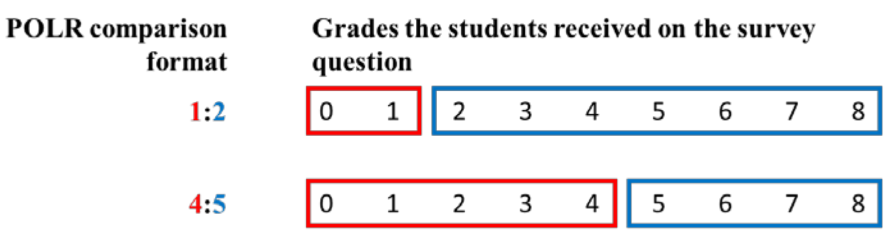


Figure 3.3: POLR comparison explanation

[13] Two of the survey questions did not have all the grade levels displayed in Figure 3.3. For the Toothbrush question, no student received a 0 grade. For the Car Design question, no student received a 0 and no undergrad received full marks (8 out of 8). While our inter-rater analysis indicated that we cannot prove that our grading scheme is not objective, further improvement of the scheme may be necessary to ensure student responses display a range of grades containing 0 and 8.

To illustrate, Equation (6) contains the values for the regression on the undergraduate responses to the transmission question comparing the grades 4 and below to the grades 5 and above (**4**:**5**).

$$\text{logit}\,(\gamma_{4:5}) = \log\left(\frac{\gamma_{4:5}}{1-\gamma_{4:5}}\right) = 2.87 + 0.17\, x_{GPA} - 1.00 x_{confidence} - 0.19 x_{GPA_{SE}} - 0.08 x_{NSE} - 0.30 x_{MF} \quad (6)$$

Equation (6) expresses that the proportional log-odds of the probability of receiving a grade of 4 or below on the boat race question as opposed to a 5 or above on the boat race question decreases by 0.19 units as x_{GPA} decreases by one unit and all other variables are held constant, and increases by 1 unit as $x_{confidence}$ increases by one unit, increases by 0.08 units as x_{NSE} increases by 1 unit, and increases by 0.19 units as $x_{GPA_{SE}}$ increases by one unit, and increases by 0.30 units if the student is male as opposed to female. The slope for $x_{confidence}$ has the largest magnitude, which means that this variable has the biggest effect on the model.

To help us understand the interaction of variables in our regression models, we need to identify how each of the variables could interact within the model to determine which ones to include. Table 3.16 contains each potential interaction term and an explanation for how it functions in the context of our models[14]. Does including these interaction terms in our model introduce too many variables to test regression against our sample size and are we at risk of overfitting the data? Peduzzi et al. [1996] state that a conservative event per variable (EPV) estimate is between 10 and 20. We received approximately 140 undergraduate student responses, and are testing against 13 variables (variables and interaction terms) in our regression analysis. This yields an EPV of approximately 10.8, which is within the conservative limits suggested by Peduzzi et al.

[14] These explanations assume that the interaction term in the model is positive. If the term is negative, the opposite of the explanation is true.

Table 3.16: Interaction term explanations

Term 1[15]	*	**Term 2**	**Explanation of Interaction**
x_{GPA}	*	$x_{confidence}$	The student only does better on the survey if they got a confidence boost from getting better grades (in general).
x_{GPA}	*	$x_{GPA_{SE}}$	Not considered because this would not make sense as an interaction term. GPA and the grades students receive in their systems courses are likely highly correlated.
x_{GPA}	*	x_{NSE}	The student only does better on the survey if they got good grades (in general) and took more systems courses.
$x_{confidence}$	*	$x_{GPA_{SE}}$	The student only does better on the survey if they got a confidence boost from getting better grades in systems courses.
$x_{confidence}$	*	x_{NSE}	The student only does better on the survey if they got a confidence boost from taking more systems courses.
x_{NSE}	*	$x_{GPA_{SE}}$	The student only does better on the survey if they got good grades in systems courses and took more systems courses.
x_{NSE}	*	x_{MF}	Male students only did better on the survey if they took more systems courses.
x_{MF}	*	$x_{GPA_{SE}}$	Male students did better on the survey when they received high grades in systems courses.
x_{MF}	*	$x_{confidence}$	Male students did better on the survey when they were confident in their systems engineering abilities.

What do the overall results of each model look like with all of the variables and interaction terms we have discussed so far? Table 3.17 contains the results of the linear and ordinal logistic regression for the undergraduate responses to the transmission survey question.

[15] The order of terms does not matter.

Table 3.17: Linear and ordinal logistic regression for undergraduate transmission responses

Variable	**Linear Regression**		**Ordinal regression**	
	Estimate ($\boldsymbol{\beta}$)	**P-Value**	**Estimate ($\boldsymbol{\beta}$)**	**P-Value**
(intercept)	22.99	0.075 .[16]		
x_{GPA}	-7.76	0.037 *	-6.80	0.055 .
$x_{confidence}$	-8.47	0.117	-7.27	0.149
$x_{GPA_{SE}}$	-1.07	0.433	-0.96	0.458
x_{NSE}	0.61	0.720	0.58	0.719
x_{MF}	9.07	0.012 *	7.86	0.012 *
$x_{GPA} * x_{confidence}$	3.25	0.036 *	2.85	0.052 .
$x_{GPA} * x_{NSE}$	-0.22	0.444	-0.21	0.418
$x_{confidence} * x_{GPA_{SE}}$	0.72	0.221	0.65	0.243
$x_{confidence} * x_{NSE}$	-0.45	0.381	-0.46	0.347
$x_{NSE} * x_{GPA_{SE}}$	0.30	0.298	0.32	0.261
$x_{NSE} * x_{MF}$	0.13	0.675	0.11	0.708
$x_{MF} * x_{GPA_{SE}}$	-0.39	0.379	-0.35	0.365
$x_{MF} * x_{confidence}$	-3.42	0.016 *	-2.93	0.017 *
0:1			-20.93	0.084 .
1:2			-18.73	0.120
2:3			-17.35	0.150
3:4			-16.35	0.175
4:5			-15.83	0.188
5:6			-14.91	0.215
6:7			-13.92	0.247
7:8			-12.27	0.307

When we discussed the examples of linear and ordinal logistics regression (equations 3 and 6, respectively), we discussed what the implications of positive and negative estimates (β values) are, as well as implications for the magnitude of these values. However, we have not yet discussed the p-value for these estimates and what this means for the variables in the model. In Table 3.17, only four p-values for variables are at least marginally statistically significant, and most are not

[16] "." Indicates marginally significant p-value (0.05< and <0.1), "*" indicates significant p-value (0.01< and <0.05)

statistically significant. Thus, we cannot make conclusions for most of the variables on how they affect survey performance for the boat race question for the undergraduate population. What variables, if any, are statistically significant for the linear or ordinal logistic regression models for any of the undergraduate survey responses? Table 3.18 contains the variables that were at least marginally significant for each of the survey questions.

Table 3.18: Significant variables for each survey question for undergraduate responses[17]

Survey Question	**Variable**	**Linear Regression**		**Ordinal regression**	
		Estimate ($\boldsymbol{\beta}$)	**P-Value**	**Estimate ($\boldsymbol{\beta}$)**	**P-Value**
Levee	$x_{NSE} * x_{GPA_{SE}}$	0.69	0.011 *	0.62	0.021 *
	$x_{NSE} * x_{MF}$	0.56	0.056 .	0.63	0.021 *
Oilrig	$x_{GPA} * x_{confidence}$	2.50	0.086 .	2.09	0.163
	$x_{NSE} * x_{GPA_{SE}}$	0.58	0.032 *	0.71	0.016 *
	$x_{MF} * x_{GPA_{SE}}$	-0.76	0.150	-0.93	0.059 .
Aircraft Maintenance	x_{GPA}	-5.14	0.057 .	-6.78	0.061 .
	$x_{confidence}$	-7.49	0.059 .	-10.30	0.062 .
	x_{MF}	-4.83	0.068 .	-7.10	0.055 .
	$x_{GPA} * x_{confidence}$	2.11	0.059 .	2.83	0.061 .
	$x_{NSE} * x_{GPA_{SE}}$	0.61	0.004 **[18]	0.79	0.007 **
	$x_{MF} * x_{confidence}$	1.95	0.064 .	2.86	0.045 *
Car Design	x_{GPA}	-5.58	0.024 *	-7.28	0.032 *
	$x_{confidence}$	-6.84	0.058 .	-8.32	0.106
	$x_{GPA} * x_{confidence}$	2.69	0.009 **	3.64	0.011 *
	$x_{GPA} * x_{NSE}$	-0.29	0.132	-0.51	0.072 .
Transmission	x_{GPA}	-7.76	0.037 *	-6.80	0.055 .
	x_{MF}	9.07	0.012 *	7.86	0.012 *
	$x_{GPA} * x_{confidence}$	3.25	0.036 *	2.85	0.052 .
	$x_{MF} * x_{confidence}$	-3.42	0.016 *	-2.93	0.017 *

[17] Some of these variables have the same level of statistical significance for both linear and ordinal logistic regression, while others are significant in one but not the other. This is because these models have different assumptions as we discussed earlier in this section.
[18] "**" indicates very significant p-value (0.001< and <0.01)

What survey questions are similar in nature and did they share any similarly statistically significant variables? Most notably, the transmission and car design survey questions both had many of the same variables statistically significant (although the transmission question has marginally significant values for these variables in its linear model)). x_{GPA} has a **negative** intercept (β) values for both its linear and ordinal logistic regression models, indicating that as this value increases, the student will receive a **lower** survey grade (linear) and the proportional log-odds of the probability of receiving a higher survey grade **decreases** (ordinal). However, the interaction term of this variable with $x_{confidence}$ is also significant, and has a **positive** intercept (β) value. This indicates that a student has to have both a high GPA **and** high self-confidence in their systems engineering ability in order to perform better on the survey.

The transmission and car design survey questions were the only questions based on personal vehicle concepts, so similar statistical results may mean that questions framed in a similar manner may test similar skills. However, is this because the responses to these questions are correlated? That is, did the students who did well on one question also do well on the others, and so the regression values we have calculated are essentially based on the same data? To test this idea, we conducted a hypothesis test for the Pearson's correlation coefficient. The resulting p-value is 5.7E-08 (less than 0.05) and the sample estimates of the correlation is 0.48 (positive, and between 0—no association and 1—perfect positive linear association), indicating that there is a moderate positive correlation between the undergraduate responses to the transmission and car design survey questions, but this correlation is not very strong. Thus, our theory is disproved and the fact that these questions have the same variables similarly significant is still of note because the questions are based on similar topics. This may be because these topics are more accessible to undergraduate students at all stages in their education because both a sophomore and a senior are likely to have at least some experience with personal vehicle designs and maintenance practices. In contrast, two survey questions are based on aircraft-related topics and do not have statistically similar variables, and this may be because performance on these questions depends on whether students have taken courses on these topics (which we cannot test because we do not have access to information on what non-systems engineering-related courses each student has taken). It may be worth exploring writing more accessible questions similar to the vehicle ones, especially because our goal is to have this survey test systems engineering ability independent of each student's major area, especially since the variables were GPA and self-confidence in systems engineering ability, which are more “general” variables that would be easy for others to measure.

The oilrig and levee survey questions had similar setups (i.e. students were tasked with using design principles to discuss the aspects and flaws of a certain design) and had one statistically significant variable in common: the x_{NSE} and $x_{GPA_{SE}}$ interaction term, indicating that a student has to have taken **many** systems engineering-related courses and **received high grades** in those courses to receive a **higher survey grade** (linear) and to have a higher proportional log-odds probability of receiving a **high survey grade** (ordinal). The aircraft maintenance question also had the same result for this interaction term, and this question had a similar setup in that the question description gave the students information about aircraft systems up-front and asked them to discuss specific aspects of each design. This result indicates that giving students a language for discussing the aspects and flaws of designs may be an accurate method of measuring their systems engineering education.

What variables were most often statistically significant? The interaction term for x_{GPA} and $x_{confidence}$ was statistically significant for the transmission, aircraft maintenance, car design, and oilrig survey questions, and all had positive estimate (β) values for both their linear and ordinal logistic regression models. This indicates that students who have high grades and high self-confidence in their systems engineering abilities are likely to do well on many survey questions.
What variables were statistically significant but had opposite estimates (i.e., positive instead of negative) for different survey questions? The x_{MF} had a positive estimate for the transmission question and a negative estimate for the aircraft maintenance question, indicating that male students performed better on the transmission question and female students performed better on the aircraft maintenance question. Additionally, the interaction term of this variable with $x_{confidence}$ also differed for these two survey questions: the estimate for this interaction term was negative for the transmission question and positive for the aircraft maintenance question. This means that female students performed better on the transmission question if they had high confidence in their systems engineering abilities, and male students performed better on the aircraft maintenance question if they had high confidence in their systems engineering abilities. Further

study into the differences between how male and female students learn and perform on test questions may be necessary to understand these results.

We have so far discussed our analyses on the undergraduate survey responses, but what about the graduate students who responded to the survey? We had to separate the responses into these two populations because a lot fewer graduate students took classes we defined as having systems engineering content, so it is difficult to analyze their responses using the variables x_{NSE} and $x_{GPA_{SE}}$. What do the linear and ordinal logistic regressions look like on the graduate student responses when we omit these variables and their interaction terms? Table 3.19 contains the variables and interaction terms that were at least marginally statistically significant for the graduate responses to each of the survey questions.

Table 3.19: Significant variables for each survey question for graduate responses

Survey Question	**Variable**	**Linear Regression**		**Ordinal regression**	
		Estimate ($\boldsymbol{\beta}$)	**P-Value**	**Estimate ($\boldsymbol{\beta}$)**	**P-Value**
Boat Race	x_{GPA}	-4.90	0.145	-5.04	0.096 .
	$x_{confidence}$	-8.84	0.113	-8.84	0.079 .
	$x_{GPA} * x_{confidence}$	2.64	0.070 .	2.66	0.046 *
Empennage	$x_{MF} * x_{confidence}$	2.92	0.121	2.96	0.061 .
Car Design	$x_{GPA} * x_{confidence}$	1.85	0.103	2.20	0.082 .
Toothbrush	$x_{MF} * x_{confidence}$	2.42	0.048 *	1.64	0.115
Transmission	x_{GPA}	-7.34	0.051 .	-5.76	0.039 *
	$x_{confidence}$	-11.17	0.066 .	-8.57	0.054 .
	$x_{GPA} * x_{confidence}$	3.18	0.050 .	2.47	0.039 *

As with the undergraduate data, the interaction term $x_{GPA} + x_{confidence}$ is statistically significant the most frequently. Additionally, each of these statistically significant terms have the same magnitude from question to question (i.e. x_{GPA} is negative for both the boat race and transmission survey questions). However, it is difficult to make conclusions on this data with so many variables omitted. We need to do further work to determine what other variables we can use to measure systems engineering ability for the graduate student responses.

3.4.2 Classification and Regression Trees (CARTs)

Another method of data analysis that provides a visualization of the data is Classification and Regression Trees (CARTs), which involves segmenting the outcomes of the data into regions based on aspects within the data [James et al., 2017]. Applied to our data, this method will show what factors are likely to combine (i.e. student classification, GPA, number of systems courses the student has taken) for a student to receive a certain grade on a survey question (see Table 3.15 for descriptions of each variable). We used the “rpart” package in R to analyze the data.

Each regression tree consists of a series of splitting rules, starting at the “top branch” of the tree. The code assigns each split based on how important each factor is at determining the grade the student received on the survey question. Different survey questions may have different factors at the top of each tree and further on through each branch. The terminal nodes or leaves of the tree describe the regions the data was broken in to. The percentages for each terminal node show the portion of the population split into each category, and R assigns shades of blue to each percentage with darker colors indicating higher percentages. Figure 3.4 displays the decision tree for the undergrad responses to the oilrig question. The most important factor, at the top of the tree, is the students’ *average grade in systems courses*, which the code split at 2.79. Students whose *average grade in systems courses* was below 2.79 are to the left and above 2.79 are to the right.

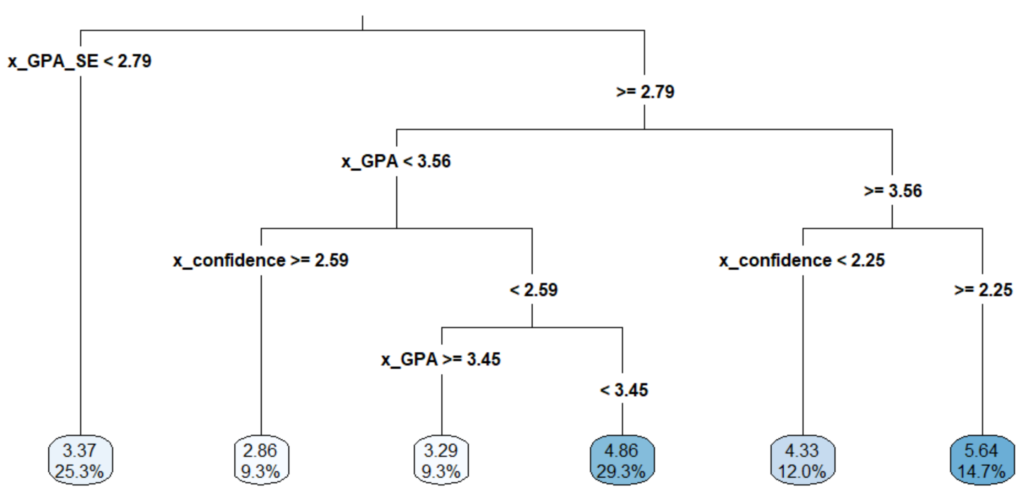


Figure 3.4: Undergraduate decision tree for oilrig question

Which variables were the most important and formed the branches for the decision tree for each survey question? Table 3.20 contains this information for each question.

Table 3.20: Variables represented in CART diagrams for each survey question

Survey Question	**Top Variable**	x_{GPA}	$x_{confidence}$	$x_{GPA_{SE}}$	x_{NSE}	x_{MF}	$x_{classification}$
Levee	$x_{GPA_{SE}}$	✓	✓	✓	✗	✗	✓
Oilrig	$x_{GPA_{SE}}$	✓	✓	✓	✗	✗	✗
Boat Race	x_{MF}	✓	✓	✓	✗	✓	✗
Aircraft Empennage	x_{MF}	✓	✗	✗	✗	✓	✗
Aircraft Maintenance	$x_{GPA_{SE}}$	✓	✓	✓	✗	✗	✗
Car Design	$x_{GPA_{SE}}$	✓	✓	✓	✗	✗	✗
Toothbrush	$x_{GPA_{SE}}$	✓	✗	✓	✗	✓	✓
Transmission repair	$x_{GPA_{SE}}$	✓	✓	✓	✗	✗	✗

6 of the 8 survey questions had $x_{GPA_{SE}}$ as their top variable, indicating that this was the most important variable in determining how the responses were split into each grade level. For 2 of the survey questions, the empennage and boat race questions, x_{MF} was their top variable. The empennage question in particular only used two of the variables, x_{MF} and x_{GPA} to split each branch. The CART diagrams for all of the survey questions used x_{GPA} to split into branches, and none used x_{NSE}. This was likely because each undergraduate had a GPA value and only a subset had taken systems courses (i.e. $x_{NSE} > 0$ for a subset of students). However, since a subset of students had taken systems courses, only some of the undergrads had a value for $x_{GPA_{SE}}$, and all of the diagrams except for the one for the empennage question used this variable to split the grades into branches. Specifically, nearly every diagram indicated that high grades in systems courses predicted high survey grades. The same is true for the $x_{confidence}$ variable, and most of the survey questions' CART diagrams predicted higher survey grades for higher confidence values.

The $x_{classification}$ variable was only used in two survey question CART diagrams (toothbrush and levee), and this indicated that seniors (students who had taken more than 7 semesters of coursework) were predicted to receive lower survey grades. We would have expected seniors to have performed better on the toothbrush question specifically because junior students (i.e. third years) take a course that has a requirements writing component that should have helped them perform better on aspects of that question.

3.4.3 Qualitative Analysis

In section 0 we discussed our grading rubric and how part of it attempts to measure systems engineering qualities that we identified in systems engineering literature. Since students whose responses received low overall scores could still have satisfied the systems engineering qualities, how do these responses compare to responses that received high overall scores? This section qualitatively describes the range of responses we received that demonstrated systems engineering qualities, using examples of survey responses. Table 3.21 displays a range of responses from two questions: flood wall and boat race, respectively. The responses have been lightly edited for typographic and spelling errors, in order to keep the focus on the content. Refer to Appendix C to see the Flood Wall and Boat Race survey questions in the manner they were presented to the students, as well as the remaining survey questions.

Table 3.21: Response range (FW: Flood Wall question; BR: Boat Race question)

Response	Overall score	Student response	Discussion
FW	Flood wall: Choose a design principle that could improve the design.		
FW-1	3	"I would say the system could absorb damage would be the most important, because no matter how much redundancy you have, if the system still fails it is not designed well."	This student thought about the "big picture" for the system.
FW-2	4	"Absorbing damage. It could also protect from large objects in the lake."	The idea of protecting from debris in flood water was not echoed in a single other student response, though this is a likely hazard in a flood.
FW-3	7	"I would choose functional redundancy to find a safe way to displace the water in addition to the wall which prevents water from flowing over the levee. Diversifying the method by which the water is safely contained would increase the number of modes by which the system would have to fail for the water to cause damage."	This response displayed a systematic approach to reducing risk to the flood protection system.
FW-4	8	"If I could choose one of the above principles to apply to this system, I would choose Absorbs Damage. It's a levee system that can tip over and completely fail if there's a flood. Not only does it fail at its purpose, but it provides local residents with a false sense of security. In some ways, using this flawed design is worse than having no levee at all, because residents will assume the levee was well-designed when in reality it was not. If the levee system was able to absorb damage, it would at least continue standing during a flood, which would greatly reduce the amount of water that is let through."	This student pointed out flaws in the system that made it more dangerous, pointing out emergent properties in the system and justifying why the design aspect they chose to improve the system was the best.
BR-#	Boat Race: consider what factors may be contributing to a boat engine failure		
BR-1	2	"The reason for boat racing: for fun, or as a job?"	The student discussed potential use-cases for the boat race but did not discuss why this was important.
BR-2	4	"The most important factor to consider is that if they don't race, they lose their sponsorship. No sponsorship means no money, and the team hasn't been winning much lately so they probably need the sponsorship money to pay employees and keep the lights on in their facilities. Regardless of whether an engine breakdown would be embarrassing, not racing could completely end the team as they go bankrupt. Even if the team can stay afloat without sponsorship money, they might have a hard time repairing the engines without any financial support. They simply have to race, regardless of whether they think they'll break down or not."	This student had a manager mindset, keeping the "big picture" goals of the team in mind and discussing why that is important.

Table 3.21 continued

BR-3	4	"Are there other options for sponsorship? (and would it be more worth it to use the rest of the season to fix issues and restart next season)"	The student thought about whether the consequences for the crew deciding to take on less risk were really that bad.
BR-4	5	"The weather as a whole. Whether there was wind gusts, inclement weather, etc. How the engine changed (if it did) after each iteration. Weighted average on what is the "most likely" temperature for which the engine would fail at."	This response took a systematic, statistics-based process to determining whether the crew should decide to race.
BR-5	7	"They should consider who the opponents of the race are and how likely they are to win with no engine failures. Try to put a money value on how devastating another loss on television would be. Hire a third party mechanic to take a look at the engine. Other weather conditions in the past can be considered too such as humidity, wind, rain, etc."	This student pointed out that the team could still lose the race even if the engine does not fail. The student also discussed having a third-party mechanic inspect the engine, which is a diplomatic way of identifying the mechanics as being part of the problem.

These responses show that students may display systems engineering concepts, processes, and ideas in their responses without necessarily also displaying specific concepts that Purdue AAE teaches in their systems courses. For example, response BR-3 considered the consequences to the crew, and response FW-2 discussed other hazards floodwall designers need to consider, each without specifically mentioning risk or hazards. Additionally, any students showed that they thought about the "big picture" in their response (e.g. FW-1, BR-2). It would be difficult to prove that these students have not learned these ideas in systems courses at Purdue.

For some responses, students took a systematic approach to analyzing the problem presented in the survey question, which may be indicative of them having learned similar approaches in their courses. The FW-3 response talked about diversifying the floodwall design to accommodate different failure modes. Additionally, the BR-4 response discussed conducting a weighted average to determine which temperatures are the most concerning for the team to consider racing at. To prove that these students likely learned these concepts in systems courses at Purdue, we would need to compare the presence of these ideas in responses to when students have taken specific courses.

Lastly, some student responses displayed ideas that are most applicable to a work environment, and the students could have learned these ideas in a classroom setting or "on-the-job" (at an internship or co-op, for example). The BR-2 response showed that the student thought as a manager of the team, keeping them motivated to think of their overall goals. Additionally, the BR-5 response framed a potential flaw in a positive manner by suggesting the team have a third-party mechanic inspect the engine, rather than placing blame on the current mechanic. It is difficult to prove where these students learned these management skills, whether it be through Purdue's AAE courses, business courses, or through on-the-job training, but it would be interesting to compare our survey response data to whether students have had co-op or internship experience. In future work we could edit the survey to ask the students to self-report their experience at the conclusion of the survey.

Each survey question received a range of responses similar to the ones described in this section. Future work will be to capture subtleties between responses that display systems engineering qualities, perhaps by further consulting the literature on systems engineering behavior or by having experienced systems engineers in industry taking our survey and comparing their responses to the students.

3.5 Interpretation of Results

Does our survey test systems engineering ability? We found that performance on questions framed using systems engineering concepts do indeed correlate with the number of systems courses an undergraduate student has taken in Purdue Aeronautics and Astronautics and their performance in those classes. We wrote two questions that presented students with design principles and had them criticize designs using these principles, as well as propose improvements for these designs using the design principles. The students who had taken more systems courses and gotten good grades in those courses received better grades on those questions than the students who did not. This could be because the systems courses taught at Purdue cover similar material, by having students analyze designs using systems engineering concepts and models.

We found that students with high overall GPA only performed better on many of the survey questions if the student also had high confidence in their systems engineering abilities. In fact, our results indicated that considering high GPA and high self-confidence in their systems engineering abilities independently resulted in both of these variables leading to poorer performance on the survey questions. Only when a student had these two aspects in concert did they do well on most of the survey questions; high confidence and low grades, as well as high grades and low confidence led to poorer performance on the survey questions. We found this result on four out of eight survey questions, ranging from questions on systems engineering concepts, to major-specific questions, to questions that covered more general technical concepts. While this strong result does not necessarily correlate to performance in systems engineering-related courses that Purdue Aeronautics and Astronautics offers, it could indicate that students are gaining confidence in their systems engineering abilities through other means (such as through co-op experience). Perhaps these students learn systems engineering concepts through internship experience or by taking courses that teach these concepts that we did not identify as having systems engineering-related material. This result could have implications for future applications of this survey: if we used this survey to vet candidates for systems engineering positions at companies, who presumably only hire candidates who have a GPA over a certain threshold, performance on the survey could indicate how much exposure the candidates have to systems engineering concepts and thus how much confidence in their abilities they have. This could be useful because a candidate may (intentionally or not) inflate their self-confidence on an application to appear as a more-qualified candidate. A high-performing, confident candidate who applies their skills and experience to many different concepts could be an invaluable resource to a company and this survey may help the company reduce their interviewing efforts to identify these candidates and thus save resources.

We also found some results indicating interactions between the students' self-confidence in systems engineering ability and their sex. We found that male students judge their systems engineering abilities more accurately than female students for topics relating to their major. Specifically, our results indicated that female students who performed well on the aircraft maintenance question rated their systems engineering abilities low, while male students who performed well on the same survey question rated their systems engineering abilities high. Female engineering students' low self-confidence is not a new problem, as described in a more than two decade-old paper by Henderson et al. [1994]. Felder et al. [1995] also corroborate our result in their study on engineering students, stating that male students rated their own problem solving abilities significantly higher than female students. The authors of this paper also discuss the societal and cultural factors that contribute to these results and propose methods for reducing these problems, such as providing students with more female role models and mentors, which may benefit the female students in the Purdue AAE program by improving their self-confidence in their systems engineering abilities. We may also want to investigate whether we framed our survey questions in a way that is less accessible for female students—does our survey itself have gender bias ingrained in it? Traxler et al. [2018] studied the Force Concept Inventory (FCI), which is used for "measuring student conceptual gains in introductory mechanics" and described how the questions' framing led to male students outperforming female students. These findings highlight that tools we use to diagnose student understanding of concepts must be studied to determine whether they introduce a gender bias.

On every question, including on questions covering general topics, students demonstrated specific systems engineering skills and abilities regardless of what score they received on the survey that our grading scheme did not necessarily measure. For example, students discussed risk analysis, value judgement models, customer needs, and use cases, which are topics covered in the systems engineering-related courses we identified. Determining whether systems courses teach students specific skills related to coursework in Purdue Aeronautics and Astronautics, such as how to use systems engineering concepts, models, and tools, may require a more thorough, pointed effort on our part. To measure these skills in some way, we may have to identify what systems engineering concepts each course teaches and analyze the students' responses to determine whether they demonstrated that they understand these concepts after taking these courses. For example, after taking a course that taught the students how to perform a risk analysis, did the students suggest to do a risk analysis on an aspect of a survey question at an appropriate time, as well as how to do so, and how it may benefit that aspect of the survey question?

3.6 Summary

In this section we described the responses from the systems engineers from Company B we interviewed gave on what discrepancies they have identified in systems engineering. We then developed 8 survey questions based on decision errors in systems engineering, how we distributed this survey, and the responses we received. We analyzed whether performance on the survey questions correlated to performance in systems engineering courses by conducting a statistical analysis on the data using the linear regression and ordinal logistic regression models. We built classification and regression tree diagrams for each survey question to visually compare how each variable branched in the data. We then described the range of systems engineering content in survey responses we received using illustrative examples. Overall, we found that the students' performance on some survey questions correlated to the number of systems engineering-related courses that students took and their performance in those courses, but these questions were framed using systems engineering concepts. We also found that high GPA and high self-confidence in the students' systems engineering ability correlated to better performance on many survey questions. Additionally, we found that female students performed well on a major-related survey question while rating their self-confidence in their systems engineering abilities poorly, which most of the male students did not do.

CHAPTER 4. HOW CAN WE HELP IMPROVE SYSTEMS ENGINEERING EFFORTS?

In this section we discuss how the information we gathered on systems engineering failures may be disseminated in a tool systems engineers may use to help them when they encounter problems on their projects. We begin by discussing criticisms systems engineers have on the systems engineering tools they currently use. Then, we describe how the network we initially built in Section 2.5 may be used to identify and provide guidance for problems systems engineers encounter on their projects. We then describe how we made this network into a web-based, interactive tool and how we determined whether this tool is useful for systems engineers.

4.1 What Criticisms Do Systems Engineers Have of Current Systems Engineering Tools?

As described in Section 2.6.3, we collected survey responses to many different types of questions in the semi-structured interview we conducted with Company B. In this section we discuss specifically the responses to questions related to tools the systems engineers use and what criticisms they have on these tools. To see all of the responses we received for each of these questions, refer to Section B.3. The questions that are relevant to this section are:

1. What are some of the general processes and tools you use?
2. Have you encountered any models or tools that you think would have been easier to use if you had learned about them in an academic setting?
3. What is a tool you find particularly useful or interesting? What strengths and flaws does it have?

For each question we will compare each response. Table 4.1 contains the responses to the question "What are some of the general processes and tools you use?" We intended this question to prompt each systems engineer to discuss the tools they use in their day-to-day activities.

Table 4.1: What tools the systems engineers use

Respondent	Response
1A	He draws the Systems "V" on his white board once a week. It's a global way of understanding where his team is in the process, but it doesn't really drive what they do. Tools he uses include: Block diagram, peer-review processes, requirements tracking and flow tools, and risk management tools.
2B	This respondent uses engineering instructions when he develops test articles. He also uses acceptance test procedures, and he specifically uses DOORS for requirements flow-down.
4A	This respondent uses functional flow block diagrams, operational sequence diagram, use-case tools, and timing analyses (complex or simple). He also uses requirements management tools for traceability and flow down of requirements and tests. More generally, he uses Matlab (and other mathematical tools) and spreadsheet tools. Some spreadsheets are huge, and he uses databases to keep track of things with large amounts of data. This is difficult to do because you can't necessarily trust the spreadsheet tool, especially when tracking document changes from multiple users.
4B	The main tool this respondent uses is DOORS, as a way to manage requirements. DOORS is very old and it is not well-integrated with other system engineering tools. A more integrated system would be a huge benefit. For instance, he does a lot of "timelines" and uses Microsoft project that lets him figure out the critical path between things. This software is not integrated with DOORS. There are way too many manual engineering processes.
5A	This respondent uses the Systems "V" a lot because it's a standard tool in the industry. The "heaviest" tool she uses is DOORS, for requirements. She also uses Risk Tool to manage risks; it's an internally generated database tool. In that tool her team manages risk for the entire program and other things like the document structure. All of the documents her team uses have to be processed through the Risk Tool. They use the Systems V as their guide to get to that point. Microsoft, PowerPoint, Excel are other tools they use. They make a ton of presentations for each of their reviews. Coordinating that is challenging.

Nearly every respondent mentioned DOORS or other requirements analysis tools. Many of the respondents also mentioned using the Systems "V" in some variation. Respondent 4A discussed that he has to use tools to manage other tools; that he does not "trust" spreadsheet tools to keep track of changes and needs to do database management for this task. While many of the respondents discussed using tools at different stages in the lifecycle process or tools for specific purposes (i.e. Respondent 2B discussed using "acceptance test procedures" and Respondent 1A discussed using risk management tools), Respondent 5A discussed how tools are used in concert to get a project from one stage to another. Respondent 4B touched upon the drawbacks of the tools he uses, specifically criticizing that the tools he uses are not integrated.

Table 4.2 contains the responses to the question "Have you encountered any models or tools that you think would have been easier to use if you had learned about them in an academic setting?" We intended this question to have each respondent think about the environment in which they learned how to use the tools they need for their day-to-day activities.

Table 4.2: Tools that would have been easier to learn in school

Respondent	**Response**
2A	None. This respondent believes that actually using a tool in a career is more useful than learning about it in school. For example, a DOORS class would not be as useful as sitting down and using this program. In school, students learn a "base language" that helps them understand the architecture of programs. However, these types of courses don't go into enough detail to help students create a DOORS architecture. Taking a class that you're not going to use in your day-to-day activity is not conducive to your time.
2B	None. He has been out of school too long to find it useful. Digital 3D modeling of parts (Catia-5) would have been useful to learn in school.
4A	Conceptual tools are relatively straightforward so the academic setting is a good place to pick up on them. Requirements traceability would have been a good one to learn in school. The implementation of the tools have some applicability for an academic setting (at the advanced level, not at excel level).
4B	No, in fact it's the opposite. Most of the tools that he learned in an academic setting he already had experience with. Generally systems engineers know what problems they need to solve and what they need to use to solve it. He doesn't think anyone should waste time using a generalized tool with no specific problem to solve.
5A	She would like to see more management coursework in school. She's found that the new engineers focus on the technical aspect but she does less technical work now (more management than technical work). New people coming in to this company get exposed to more project management tasks now. Project management training would be extremely useful (concept to delivery type learning). Some of the programs she works with have new hires doing more project management (even fresh out of school) She says a lot of people on her level are groomed to be chief engineers and program management.

Since we interviewed experienced systems engineers, some of them had graduated too long ago for an education on systems engineering tools to be useful for them now; they may have forgotten it by now and the information on tools that long ago would likely be out of date by now.

Respondent 2B had this observation in particular, and suggested that training in 3D modeling software, which is a component of most engineering programs today, would have been useful for him. Some of the respondents suggested that education on DOORS would be useful since it is such an industry-wide tool, although Respondent 2A said that it is easier and faster to learn DOORS in-context, on the job, which was echoed by Respondent 4B. In that same vein, Respondent 4A distinguished between different types of tools, saying that "conceptual tools" are easier to learn in an academic setting. Respondent 5A criticized that systems engineers do not receive any training on project management tools, which she said are important because systems engineers often get promoted to management roles.

Table 4.3 contains the responses to the questions "What is a tool you find particularly useful or interesting? What strengths and flaws does it have?" We intended this question to have the systems engineers discuss the strengths and flaws of the tools they use so that we could potentially use this feedback when developing our tool.

Table 4.3: Tools the systems engineers find useful

Respondent	Response
1A	The internally generated database has an inconsistent use between commercial and military programs. Someone coming from the military side can easily learn how the commercial side uses the tool, and vice versa. But there is not one way to use the tool. Users have to spend time learning if they go from military to commercial or vice-versa. Commercial has a high number of customers that have few aircraft. They are all tracked under one contract. They do things to improve the product, paid for by this company. Military has few customers with a high number of aircraft. They don't do anything the customer doesn't pay for. He has network latency issues and interface issues through his personal computer. Sometimes information is hard to find. There is tribal knowledge about how to make the tool work.
1B	The quantitative analysis tool is an Excel spreadsheet running on the user's local laptop that sometimes slows the laptop greatly. It should be hosted on a more capable device, like a company server.
2A	DOORS is a useful tool because it has co-location of material. A user can export out to other formats, and this company has done an excellent job of generating specific scripts or tools that work with DOORS that allow one to create things that it wouldn't be able to do on its own (e.g. exporting out to Word document). This company has a lot of these tools that enhance the use of DOORS as a whole. Even if you don't have 100% of the team working in the tool itself, you can export information out to Word and allow for crosstalk. Additionally, their customer has full read access to DOORS so they can see everything. It would be better if everyone was trained to use DOORS, but they can get from A to B with little difficulty. Customers specify in their contracts that they want full read access but don't necessarily use it. You have to get people to actually use it.

Table 4.3 continued

2B	This respondent agrees that DOORS is a useful tool. He wishes everything he did relating to requirements and verification and documentation was done in DOORS. Not everyone is familiar with DOORS. It would be a single place to input information. Half the people know how to use it. So you export out of DOORS and adds more manual steps. If it was used entirely you would have documents, test plans, and test reports in DOORS.
3B	DOORS is a useful tool. However, the tool is not the most user-friendly and a little overwhelming, but users get used to the flaws over time. There are intricacies in linking models to keep traceability of requirements. Traceability becomes cumbersome. He’s not sure if this company has tried to find something better (i.e. more user friendly).

The systems engineers criticized the tools they use in different ways: usability, inconsistent use, and incompatibility with their personal computers. Regarding usability, the systems engineers had criticisms on features of the tools they use (e.g. Respondent 3B said that DOORS is “not user-friendly”, and multiple respondents said that many of the tools they use require time to become familiar with them). Some of the respondents said that the tools they use are (1) not used by all of the people who work with them on their projects and (2) not consistently used across the company. Respondent 2A said that a good feature of DOORS is that information from the tool can be exported to other programs so that people on the project can still get the information. This respondent also mentioned that the tools need to be compatible with how their customers use them. Respondent 1A discussed that a tool he uses is used differently by systems engineers depending on what kind of project they work on: military or commercial applications. Finally, some of the systems engineers criticized that the tools they use are frustrating because they are incompatible with their personal computers and require long load times.

We learned from interviewing these systems engineers about the tools they use. We found that many of these engineers use tools related to systems engineering tasks, such as requirements management and risk management. Many of these engineers stated that the tools they use could be more user-friendly, and should interface with their computer equipment and other programs more seamlessly. The engineers also expressed frustrations regarding how the tools are used throughout their programs; since not everyone uses or is required to use the tool, it is not as effective for their program as a whole. Many of the engineers stated that learning about the tools they use in an academic setting is not necessary.

4.2 Application of the Cause-Recommendation Network

Can we build a systems engineering tool that disseminates the information we found in our study of systems engineering failures and incorporates some of the feedback we received from systems engineers on the tools they use? Suggestions for improving on problems in systems engineering failures are often either so general they are essentially platitudes ("put your best people on the job"), or highly specific to particular contexts (e.g., "replace the faulty burst valve"). In contrast, because the recommendations we identified are based on a set of actual accidents, they can be ranked in terms of frequency, and each one can be traced back to one or more accidents, thus providing concrete examples of what went wrong as well as context for each recommendation. Here, we demonstrate two aspects of how the information in the cause-recommendation network can be used to identify useful and informative guidance.

4.2.1 Identifying and understanding potential causes

An organization that suspects it may have problems can use the network to identify the most frequent causes. The most frequent cause in both accidents and project failures is *failed to consider design aspect* (Figure 2.6). To help illustrate this and the other causes, the network also provides over 900 "back stories" that summarize how each cause manifested in our case studies. Table 4.4 shows examples of these back stories from both accidents and project failures for *failed to consider design aspect*.

Table 4.4: Back stories for *failed to consider design aspect*

Failure	Cause Back Story
Upper Big Branch Mine explosion [McAteer, 2011]	The Upper Big Branch Mine was a coal mine in West Virginia that suffered an explosion that killed 29 miners. Coal mines require constant "rock dusting" to keep coal dust levels down to prevent explosive atmospheres from forming within the mine. Among other causes, the mine was so large that workers conducting rock dusting had to make many trips to reload material to rock dust the entire mine. **Design aspect not considered:** A chute-like delivery system to the center of the working area of the mine would have made rock dusting easier.
ValuJet flight 592 crash [NTSB, 1996]	Contractors working for ValuJet Airlines were refurbishing an aircraft and removed its expired chemical oxygen generators, used to supply oxygen to passengers in situations when a plane suffers a decompression during flight. The contractors improperly packaged and labeled the generators as empty rather than expired. Eventually the expired, but not empty, generators were shipped on flight 592. During takeoff, a fire started in the cargo hold, and would have burned itself out had the (now damaged) generators not supplied the fire with oxygen. The plane was eventually overwhelmed by the fire and crashed. The passengers and crew were all killed on impact. The NTSB report also noted that even if the aircraft had managed to land, the passengers might have been injured or killed by toxic air. **Design aspect not considered:** The emergency oxygen masks deployed during in-flight emergencies do not separate cabin air, which could be toxic in the event of a fire, from the oxygen flow.
Westray Mine collapse [Richard, 1997]	The Westray Mine was a coal mine in Nova Scotia that had a history of problems because the mine's management frequently took shortcuts to improve production at the cost of safety. **Design aspect not considered:** the ventilation system in the mine was designed in a haphazard way; for example, the fans were placed in locations within the mine that were not conducive for the air flow. Thus, the ventilation system allowed methane gas and coal dust to build up, eventually causing the mine to collapse.
Iridium satellite phone cancellation [Mcintosh, 1999]	The Iridium satellite phone was a phone that could connect a call anywhere on Earth at a time when cell phone coverage was unreliable. The phone did not sell well, and the founding company declared bankruptcy, although the satellite system remains operational. **Design aspect not considered:** Designers did not properly consider their customers' needs. The phone was extremely expensive, calls could only be made outside (within line-of-sight of the satellite network), the phone was difficult to use and required special training, a special cartridge was required to make conventional mobile network calls, and the phone itself was large, weighing over 1 lb.
Seawolf Navy Submarine delays and cost overrun [GAO, 1994]	The Seawolf Navy submarine was delayed and over-budget. **Design aspect not considered:** Two contractors who had originally competed to win the contract were commissioned to design and build the aft and forward sections of the submarine separately. This decision underestimated the immense coordination and cooperation that would be required between the contractors, as well as the extensive design and construction rework the program eventually needed.
F-35 Lightning II delays and cost overrun [Ciralsky, 2013]	The F-35 Lightning II is currently delayed and over-budget. **Design aspect not considered:** The aircraft is intended to be one-size-fits-all for the United States Navy, Air Force, and Marines, which means that a single design, with slight modifications, is meant to meet the needs of all three customers. This common platform decision did not fully consider the challenges and compromises involved in trying to meet divergent needs. In addition, development on all three variants is delayed whenever a common part fails.

4.2.2 Identifying and understanding potential recommendations

Figure 4.1 shows the 16 recommendations, ranked by the percentage of accident causes connected to each one. The percentages do not add up to 100% because many causes are linked to more than one recommendation code (see Figure 2.8) and some causes are not linked to any recommendations. For example, *make instructions more clear* accompanied 17% of the causes in accidents that had recommendations. An organization seeking to make general improvements without prior knowledge of problems should start by following the recommendation codes with the highest percentages. These recommendations are most likely, based on our dataset, to be applicable in any given organization.

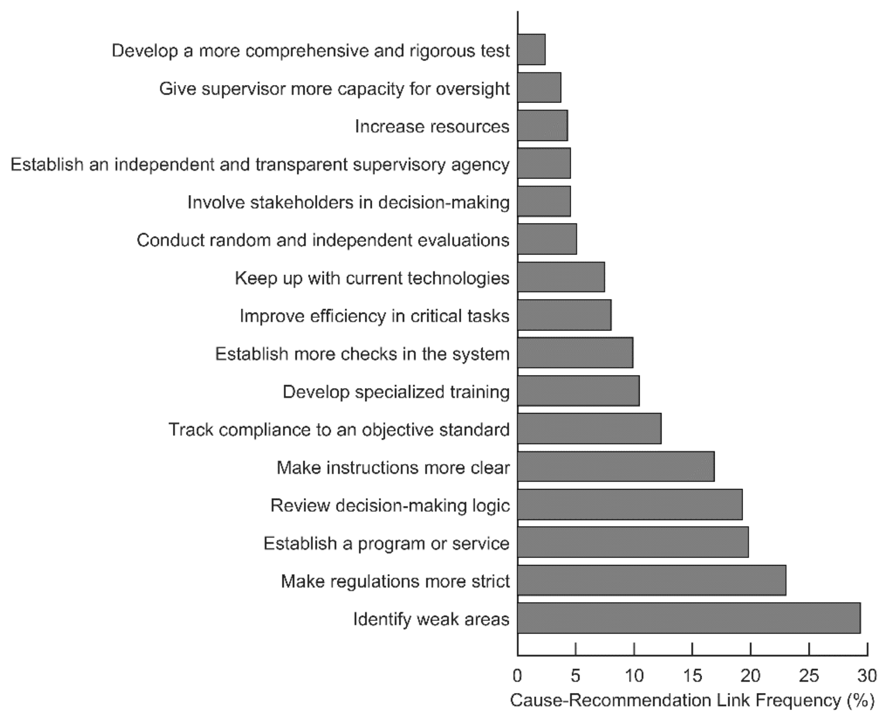


Figure 4.1: Recommendations ranked by cause-recommendation link frequency in accidents.

In Figure 4.1, It is not surprising that *identify weak areas* was most often recommended—it is hard to imagine a scenario in which identifying weak areas is not a good idea! Similarly, many of the other recommendations also appear self-evident, but may be hard to translate into concrete context-specific terms. To help address this problem, the cause-recommendation network provides 600 back stories of the recommendations and the problems that led to the recommendations. For example, Table 4.5 shows examples of why and how investigators made the recommendation *identify weak areas*, which appears in 25 out of 30 accident investigations and is linked to 29% of accident causes.

Table 4.5: Examples of source accidents for *identify weak areas*

Code: Identify weak areas (29%): Assess what aspect of the system is potentially neglected.		
Cause	**Accident**	**Recommendation**
Failed to consider design aspect	Swissair Flight 111 Crash [TSBC, 1998]	A fire started on the aircraft while it was flying, and because it propagated in unoccupied parts of the aircraft, it went unnoticed and eventually brought the plane down. Investigators found that the air filtration system in the aircraft was designed in a way that the filters removed smoke from the air and so no one knew about the fire until it was too late. The investigators recommended that all aircraft systems be evaluated and re-designed to that they do not make fires worse once they have started.
Conducted maintenance poorly	Deepwater Horizon oilrig blowout and fire [CSB, 2014]	The Deepwater Horizon oilrig was plugging a new oil well in a standard procedure. The well's plug burst and the ensuing oil spurting from the well caught on fire and destroyed the rig. Investigators found that a component designed to pinch the drill pipe shut in the event of a blowout was unreliable and did not close the pipe after the plug burst. The Chemical Safety Board recommended that operators identify safety critical elements and prioritize them for inspections and maintenance.
Failed to inspect	Aloha 243 crash [NTSB, 1989a]	Aloha Airlines was flying their B737 aircraft on routes between the Hawaiian Islands, which is a very warm and humid environment. Personnel conducting maintenance inspections missed evidence of corrosion on the aircraft body. As a result, the weakened fuselage gave way to fatigue, and the cabin explosively decompressed. The NTSB recommended that air carriers frequently assess the performance of their maintenance departments so they can improve their inspection practices and better identify corrosion damage on aircraft.
Managed Risk Poorly	Buncefield oil fire [Newton, 2008]	The Buncefield oil storage depot was filling a storage tank with oil when a gauge designed to detect when the oil reached a high point failed. There was no alarm and the receiving site could not halt the flowing oil. The tank overfilled and a spark lit the spewing oil on fire, causing an explosion. The gauge had stuck many times in the past, but personnel merely un-stuck the gauge each time, rather than fixing or replacing the gauge. The investigators recommended that the oil storage company investigate root causes of failures in safety critical elements at the plant.
Failed to supervise	Colgan 3407 crash [NTSB, 2009]	Colgan Air flight 3407 was on approach to Buffalo in icing conditions, and the pilot had the aircraft on autopilot, which made it more difficult for him or the co-pilot to realize that the wings were icing. The autopilot extended the flaps for landing, and the aircraft went into severe pitch oscillations because of the ice on the wing. The pilot took manual control of the plane, and, as the plane began to stall, the pilot overrode many safety precautions designed to prevent a stall, which caused the plane to crash. The NTSB found that the captain had training and proficiency deficiencies that Colgan had not addressed, and recommended that all air carriers identify pilots with similar deficiencies so they can improve their performance.

Table 4.5 continued

Created inadequate procedures	Xcel Energy explosion [CSB, 2007]	The Xcel Energy hydroelectric plant had underground pipes with deteriorating coating that had to be replaced. As part of replacing the coating, workers used a flammable solvent to keep the epoxy sprayer from clogging. Among other causes, the procedures for the coating process did not specify how much solvent could be used safely in a confined space. The personnel used too much solvent for the small space and created an explosive environment, which ignited and killed the workers in the pipe. The Chemical Safety Board recommended that organizations identify and control ignition sources and monitor confined spaces when using flammable materials.

If an organization has identified a particular problem behavior, it can use the cause-recommendation network to identify the most appropriate recommendations for addressing that behavior. For example, suppose an organization discovers that it did not adequately supervise a project. Table 4.6 shows the associated recommendations for *failed to supervise*, as well as the relative ranking of each recommendation, based on how often we connected them to *failed to supervise*, described in percentage as well as raw count. Thus, for example, *identify weak areas* was recommended 16 times in response to *failed to supervise*, which we identified a total of 117 times in our accidents and project failures. Thus its percentage is $16/117 \approx 14\%$.

Table 4.6: Recommendation codes linked to *failed to supervise*

Recommendation Code	**%**	**#**
Identify weak areas	14%	16
No recommendation code	14%	16
Establish an independent and transparent supervisory agency	11%	13
Establish a program or service	10%	12
Conduct random and independent evaluations	9%	11
Make regulations more strict	8%	9
Increase resources	7%	8
Review decision-making logic	5%	6
Track compliance to an objective standard	5%	6
Develop specialized training	4%	5
Make instructions more clear	4%	5
Give supervisor more capacity for oversight	3%	4
Involve stakeholders in decision-making	2%	2
Keep up with current technologies	2%	2
Establish more checks in the system	1%	1

Table 4.6 continued

Improve efficiency in critical tasks	1%	1
Develop a more comprehensive and rigorous test	0%	0
Total	**100%**	**138**

The network also allows users to sort by other categories, such as industry type—a user could, for instance, see all causes related to government acquisitions or aircraft crashes.

4.3 Cause-Recommendation Tool Development

To make our cause-recommendation network in an interactive, easily accessible form, we first developed a prototype of the network using Tableau. We then used the prototype network to commission the final interactive network, which was created using JavaScript. Finally, we conducted usability and usefulness testing on the final design.

4.3.1 Prototype Development

We constructed our prototype network in Tableau, a popular interactive modeling tool, to enable user interaction with the network. We chose Tableau because it is an easy-to-use data visualization tool (the software "helps nontechnical people translate large data sets into stunning visuals with a simple drag-and-drop interface" [Solomon, 2016]). Building our network with an easy-to-use interface is important to us because we want to be able to build the prototype quickly and easily so we would be able to define the needs for the network for a contractor to create in a different format. Tableau is also compatible with the other software we used to analyze our data, like Excel (where we stored the data) and Matlab (how we initially analyzed the data and what we used to construct the network in Figure 2.12). Finally, Tableau allows its users to publish their work to the internet with ease, and offers free workbook hosting on Tableau Public. Our upfront costs for creating the prototype were thus minimized.

After constructing our tool in the Tableau platform, we encountered problems with the platform, such as baked-in rigidity that kept us from fully customizing the network and allowing functions like letting the user click on nodes in the network and interact with it in other ways. For example, it was difficult to perform simple tasks like moving node labels and creating label leader lines to make the network clearer. Additionally, some of the stories were verbose and require more space than Tableau allots, which means that the user is not able to read the full story. Lastly, the tool required long loading times and did not provide adequate feedback to the user on its functions (recall that the systems engineers had similar complaints for some of the tools they use in Section 4.1). For these reasons, we decided to construct the final version of our tool using a different platform.

4.3.2 Full-scale Tool Development

We constructed our interactive solution aid using HTML and JavaScript (available online at https://engineering.purdue.edu/VRSS/research/force-graph/index_html) which eliminated many of the problems the prototype network experienced. For instance, the load time is shorter and we were able to customize features of the tool to a greater extent. The tool is easily edited to include more cause-recommendation stories, systems engineering failures, causal actions, or actors as our research progresses. Figure 4.2 shows how our network appears when the user first navigates to it. In the figure, the network is visible to the left, with all of the causes and recommendation codes displayed with their connections, as in Figure 1. On the bottom left, the user may use an expanding arrow to view the descriptions of the causes and recommendations. On the bottom right are the accident stories. The user may use the expanding arrow to display the stories for each accident, sorted by industry (e.g. aerospace, energy, transportation), then sorted alphabetically. The solution aid displays how many stories will appear if the user opens all of the expanding arrows. To the right of the network the solution aid displays a randomly-selected story from the list, which changes if the user interacts with the tool or reloads the webpage. Along the top of the figure are three drop-down menus, which the user may use to toggle between networks (cause-recommendation network, accident causes, and project failure causes), people involved in the failure (e.g. designers or operations managers), and industry (e.g. aerospace, energy, or transportation).

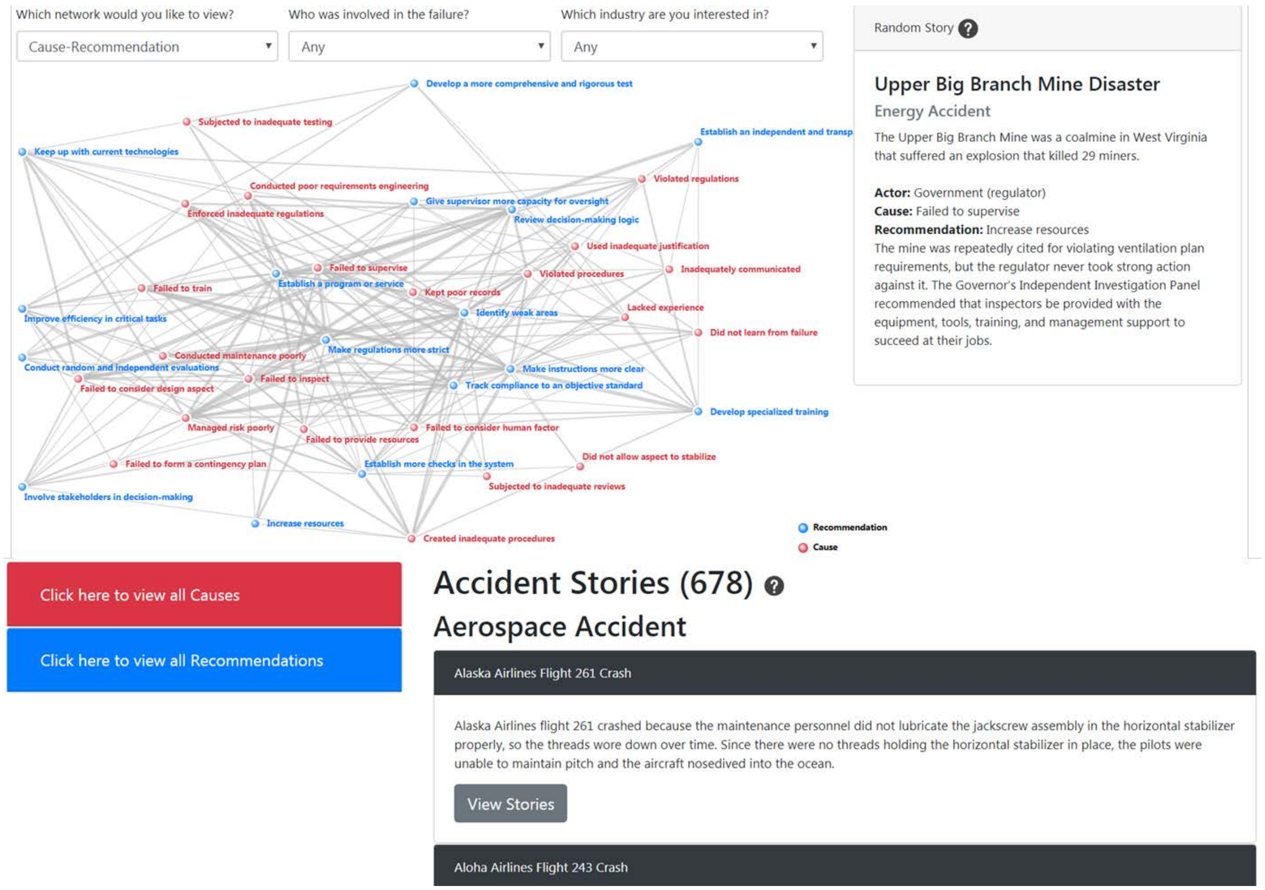


Figure 4.2: Interactive solution aid as it first appears to the user

4.3.3 Usability Analysis

To determine whether users would encounter errors in the network's interface, we conducted usability testing. We had subjects use the network while filling out a short survey, which asked the subjects questions on the look and feel of the network. The survey also asked the users to use the network from the point of view of company representatives experiencing problems on an imaginary project. Appendix E contains the usability survey in its entirety. We also consulted with a usability and user experience expert on the state of our network, who also suggested we use the information in Krug [2009] on usability. Table 4.7 contains specific feedback from our usability study and the usability expert and the changes we made based on that feedback. For the full usability survey, refer to Appendix E.

Table 4.7: Changes to the network based on usability input

Criticism on the network from usability study and/or usability expert	**Change we implemented in the network**
Users did not realize that the nodes on the network were interact-able	We made the nodes more button-like to encourage the user to click on them
The network displayed a lot of information at once and overwhelmed the user	We included more "click to reveal" functionality where the user has to interact with the tool to reveal information on it, such as clicking on a box to reveal all of the cause definitions. Additionally, we consolidated the stories so that the overall summary of each failure is only displayed once, making the stories less repetitive.
The network did not provide feedback when the user interacted with it	We made the network more responsive to user input, including: (1) having a "random story" at the top of the page that changes when the user interacts with the network, (2) having the definition of the node the user clicks on appear at the top of the list of cause/recommendation definitions, and (3) displaying the number of stories relating to that node at the top of the Accident (or Project Failure) Stories list.
The network was difficult to understand	We made the network easier to understand by (1) changing the colors we used on the network, (2) displaying a key describing red nodes as causes and blue nodes as recommendations, and (3) having information bubbles next to items users identified as requiring more descriptions.

We implemented these changes to the network and improved its usability so that the users could concentrate on the network's content and allow it to help them with their problems, rather than spending their time struggling with the network's interface.

4.3.4 How to Use the Tool

This section describes the interactive functions of the solution aid and how it can be used to help organizations experiencing problems to learn from our massive dataset.

The “Accident Stories” section of the solution aid displays all of the stories that comprise the network, sorted by industry (e.g. aerospace, energy, transportation), then sorted alphabetically. The user may filter these stories by clicking on nodes in the network (red for causes and blue for recommendations)—so that only stories associated with that cause or recommendation appear. Figure 4.3 shows this filtered view of the stories. In this example, when the user selects the cause *conducted poor requirements engineering*, the *Accident Stories* window displays only stories that have to do with this cause. Note that the definition of the selected node is displayed under the network, and that the number of accident stories displayed has changed to reflect how many stories contain *poor requirements engineering*.

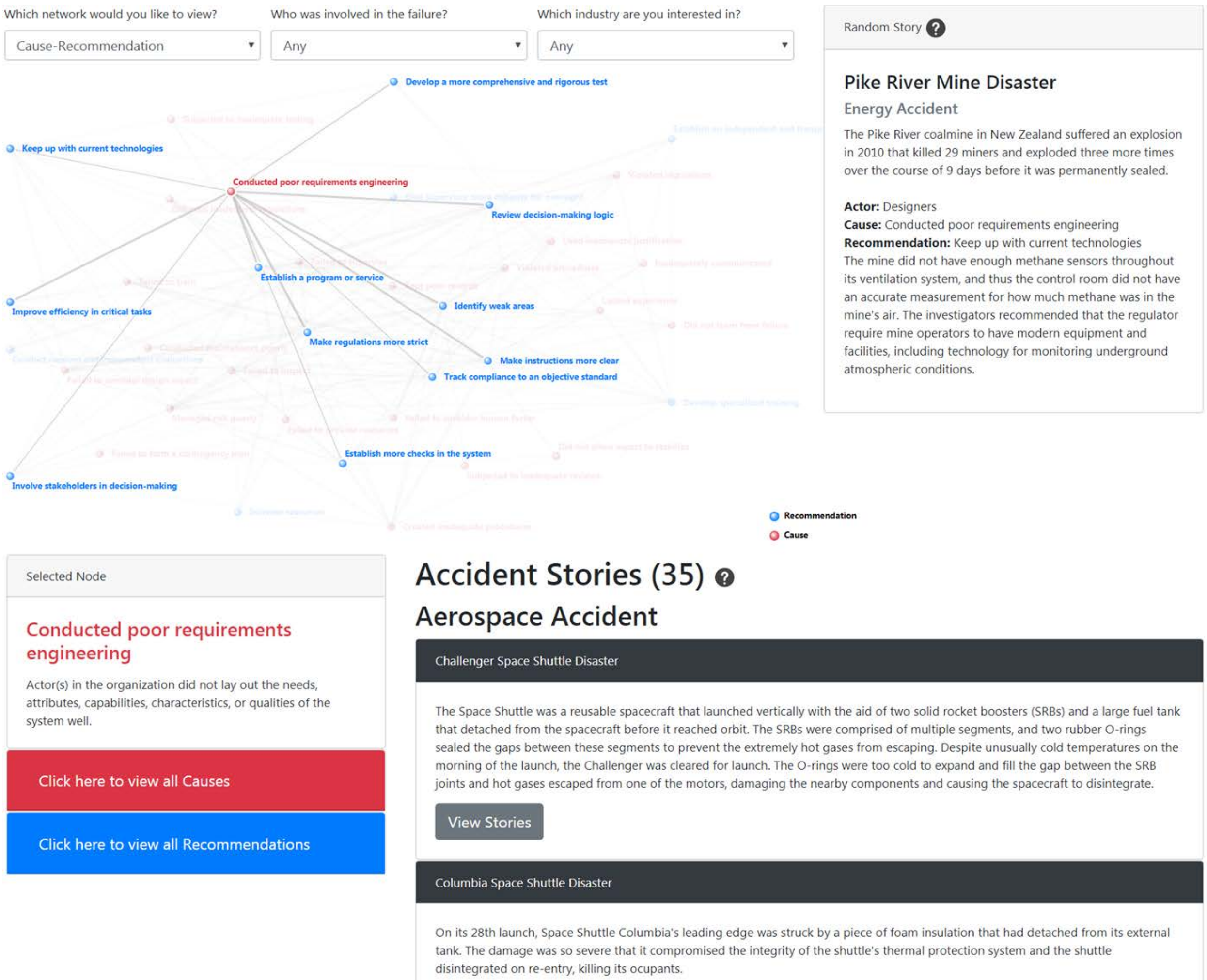


Figure 4.3: User node selection example

To view the stories under each accident, the user may click on the button *View Stories*. To view the definitions of the causes and recommendations, the user may click on the rectangles labeled *Click here to view all Causes* and *Click here to view all Recommendations* respectively. Figure 4.4 shows the solution aid with the expanded list of Challenger Space Shuttle accident stories, as well as the expanded list of cause definitions.

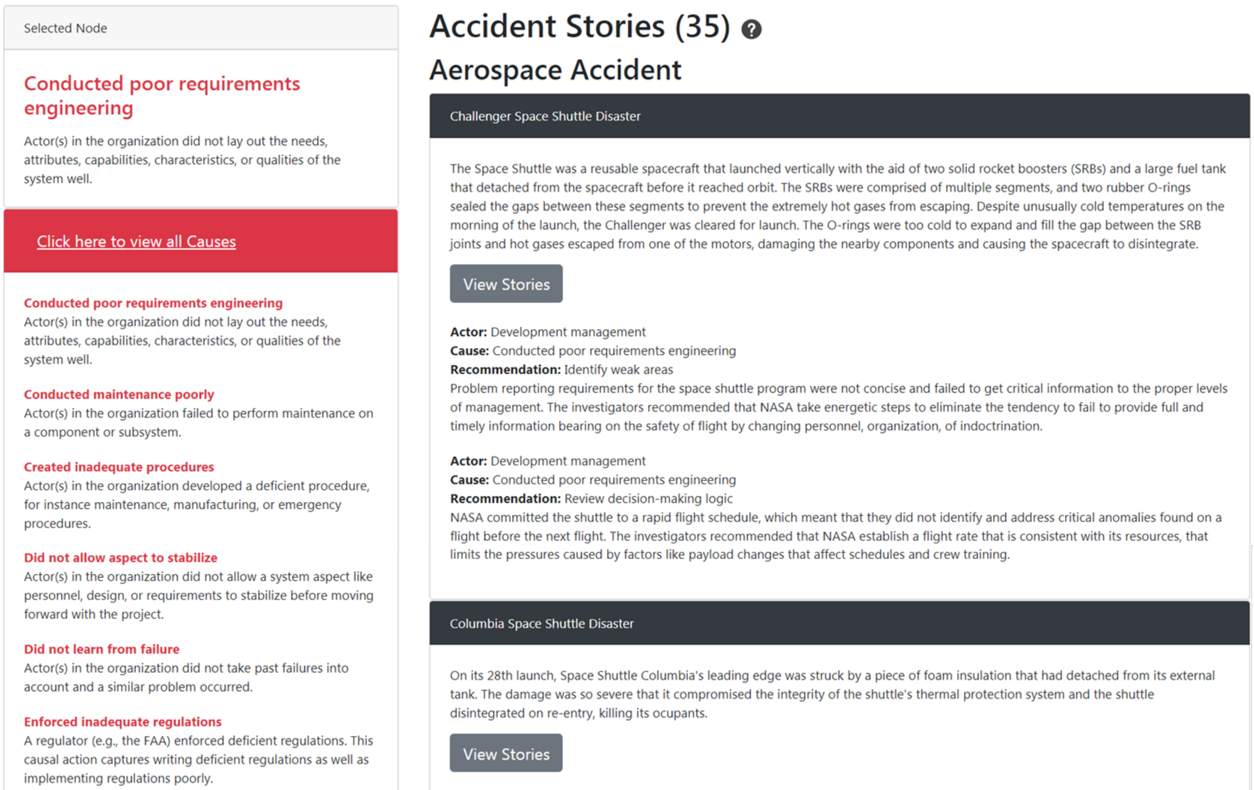


Figure 4.4: Expanded stories and nodes view example

As stated earlier, the user may use the drop-down menus along the top of the solution aid to toggle between different views. Figure 4.5 shows how the network appears if the user selects *designers* under the *Who was involved in the failure?* drop down menu. The network and the accident stories now display only the causes (and recommendations connected to them) that involved designers. The network will change depending on which item is selected in each drop-down menu.

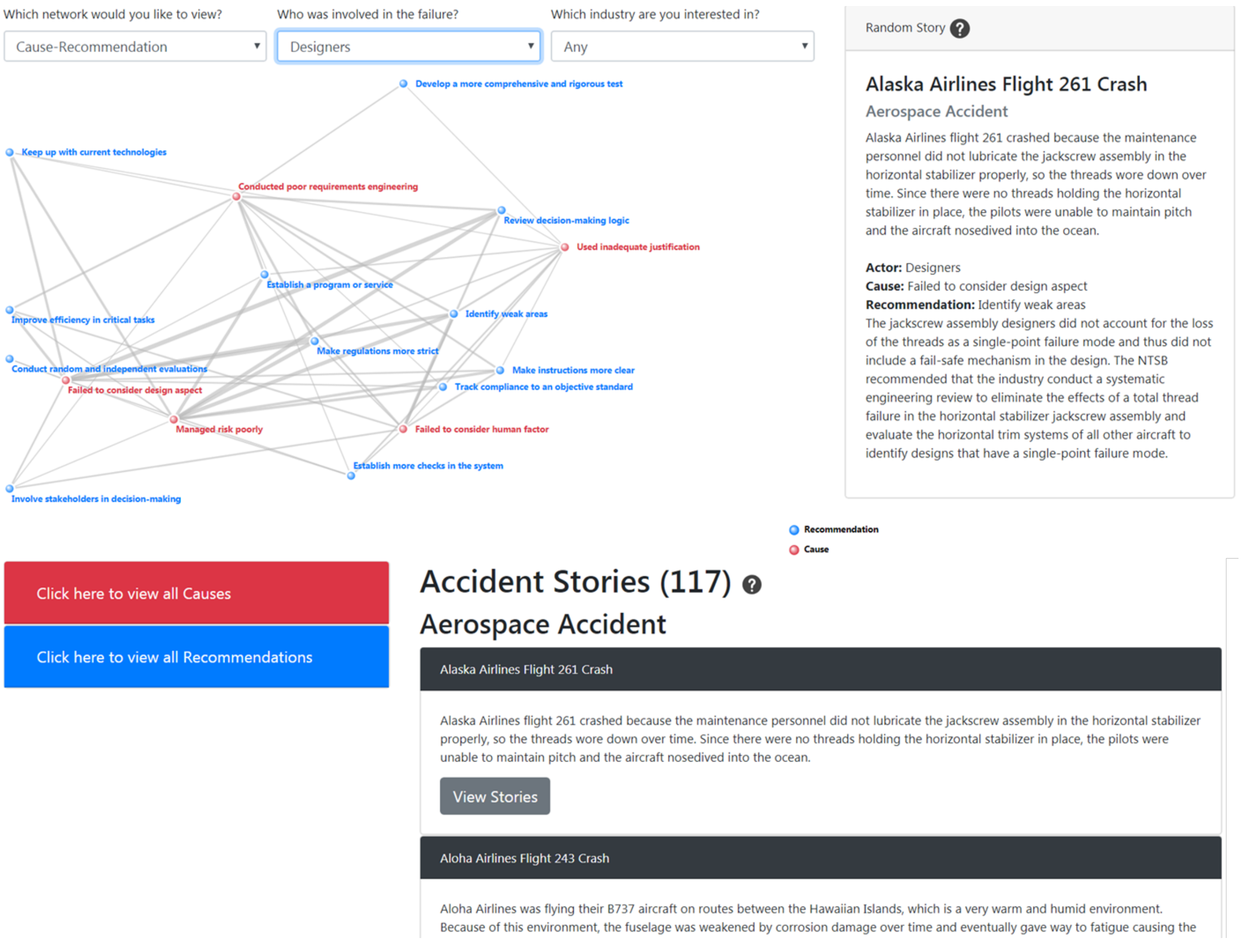


Figure 4.5: Network sorted by actor example

4.4 Is the tool useful?

Is our tool useful for systems engineers trying to form remediation measures for problems they encounter on projects? To investigate this question, we developed a set of surveys based on two NASA failures, in which we extracted findings from these reports (similar to the method we used to construct the tool) and presented them to interviewees so they could propose what they would do to solve these problems. We selected both failures in aerospace fields so that the interviewees would have some familiarity with the topics presented. We anticipated that most of the people we interviewed would have aviation backgrounds (and have not worked on space applications) so we deliberately selected one aviation-related failure and one space-related failure to investigate what the interviewees would do when faced with a more unfamiliar scenario. Would the tool help the interviewees more with this unfamiliar scenario since it is comprised of a wide variety of failures? Figure 4.6 shows a diagram of our survey scenarios: each one contains some combination of two failures (e.g. scenario D and scenario A, or DA), but does not repeat the same NASA failure in the same combination (e.g. scenario AC or BD).

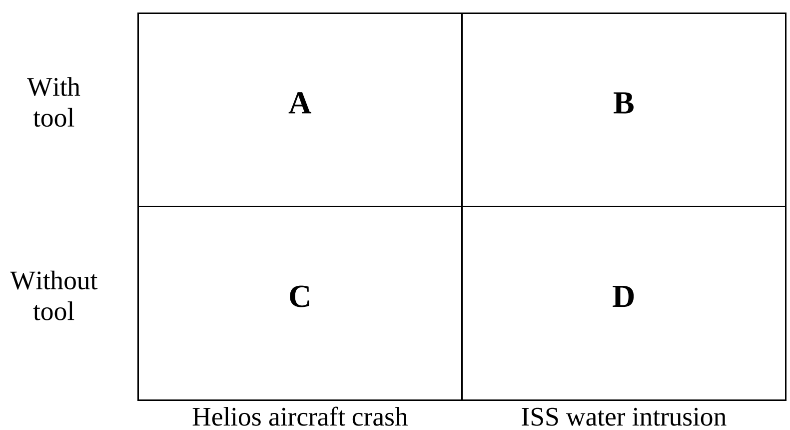


Figure 4.6: Usefulness study design

Since we wanted to gauge the change in the representative's performance when using the tool, we designed the surveys so that some scenarios involved answering questions on a NASA failure without the tool, then answering questions on a NASA failure with the tool (e.g., scenarios DA and CB), while others involved answering questions on both NASA failures without the tool (e.g., scenarios CD and DC). This is so we can compare any performance improvements that may be the result of having the aid of the tool with performance improvements that may be the result of becoming more familiar with the survey format. Appendix F describes the NASA failures we selected for the survey scenarios, the findings we selected to present to the interviewees in the surveys, and the questions we asked. At the conclusion of each interview we asked the participants who did not use the tool while answering questions on the NASA failure findings (e.g., scenarios CD and DC) to go through the tool and answer general questions on it to give us additional feedback.

This type of study design is also referred to as a nested design, in which one factor B is nested in levels of another factor A [The Pennsylvania State University, 2018]. For our study, factor A is the "preparatory failure", or the first failure the interviewee answers questions on. Factor B is then whether the interviewee uses the tool or not while answering questions on the second failure. Figure 4.7 shows a visual representation of our study in the nested design format.

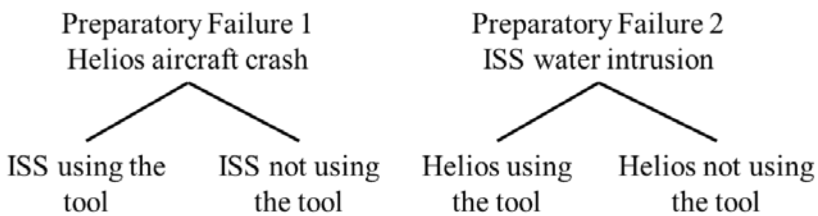


Figure 4.7: Nested design representation

We interviewed four industry representatives at a large-scale aerospace company (Company B let us come back and interview their systems engineers). Some of these systems engineers suggested that this cause-recommendation tool would be more useful for people less experienced in systems engineering, so we also interviewed 17 "novices" in systems engineering using the same survey formats. In this section we first describe our groups of experts and novices and their responses at a high level, then compare their responses on detailed quantitative levels. We then discuss what the implications are for our cause-recommendation tool.

4.4.1 Interviewee Description

We met with four industry representatives at a large-scale aerospace company, and randomly assigned them one of each of the four scenarios. Each representative had decades of systems engineering experience and had worked at that company for a majority of their systems engineering careers. We thus considered them "experts" in systems engineering. The experts required more time than we anticipated to get comfortable in the interview environment by taking their time to answer lead-in questions on their background and the types of projects they worked on. We were thus not able to complete 3 out of 4 interviews within the time allotted. Since the interviewees who answered questions using the tool did so at the end of the interview, it meant that we did not receive as much data with the experts as we wanted on using the tool.

Subsequently, we also interviewed 17 relative "novices" in systems engineering: graduate students in Purdue Aeronautics and Astronautics who had some experience in systems engineering. We defined different ways that these students could have accumulated experience in systems engineering, including (1) majoring in systems engineering as part of their degree program; (2) taking systems engineering courses, like senior design or design-build-test classes, whether they were offered at Purdue or otherwise; (3) having work experience, either internships or full-time, measured in months; and (4) self-defined systems engineering experience that does not fit within these categories, such as a research project that had aspects of systems engineering. The novices were under more loose time constraints than the experts, so each one answered all of the questions on the survey.

4.4.2 High-level Comparison

How do the expert and novice responses compare at a high level, and are these findings corroborated by literature comparing expert-novice behavior? In this section we describe differences between the expert and novice recommendations and tool usage. Since we received far more novice responses than expert responses (17 novice and 4 expert), it is difficult to make concrete conclusions about the differences between the two groups. In this section we discuss the responses generally and frame them using references that discuss expert and novice behavior.

In general, most of the experts and novices were more comfortable with the Helios failure than the ISS Water Intrusion failure because most of them had more experience with aircraft than space systems, except for two novices majoring in astrodynamics. For most of the novices, this led to higher reliance on the tool when they answered questions on this failure and had access to the tool (the expert who did this interview scenario did not have enough time to complete all the questions on this failure while using the tool).

One observation of our interviews was that the experienced systems engineers did not seem to distinguish between subtleties in the findings. Once each representative made an initial recommendation within the same NASA failure, they kept returning to this recommendation for the subsequent findings in that failure. One expert responded for all five findings for the Helios failure that the engineers should have used publicly-available guidelines on airworthiness when designing the aircraft (although using slightly different language each time). The novices seemed to distinguish between subtleties in the findings better and rarely repeated recommendations from one finding to the next.

This result on expert-novice behavior is corroborated by Kim & Ryu [2014], who found that experts are effective at framing design problems, make decisions quickly, and are more wedded to their own previously developed design concepts than novices. This result may also be explained by experts being more adept at pattern recognition in their field [Bilalić et al., 2010] and subsequently using this skill to ignore subtleties to formulate an appropriate response that fits their experience of problems they have seen before. Another possible explanation is that experts are better able to recognize underlying principles, rather than focusing on the surface features of a problem [Cross, 2004]. Perhaps the expert saw their initial recommendation as solving the underlying problem, which would have worked for all of the findings from the NASA report, regardless of the subtleties in the findings.

We intentionally chose an array of findings from each failure so that some were more well-defined than others. In the Helios aircraft crash, for example, we chose findings related to more general engineering tasks, like failing to use robust models to verify the aircraft's operating environment, as well as tasks that are more specific to systems engineering, like not doing hazards analyses properly. One finding from that report, however, was a bit of a departure from either of these problems that have been well-tread for most engineers, but was similar to other problems we found in our study of past systems engineering failures. This finding criticized the "team dynamics" of the Helios flight crew during the test flight, and said that the crew's management had not clearly defined the members' roles and responsibilities, which led to people not speaking up when they saw problems with the aircraft. When one expert read this finding, he did not offer a remediation measure because he claimed this was not a problem directly relating to systems engineering and that systems engineers would likely not have been present at this stage in the project's lifecycle. When faced with this finding, novices who did not have the tool at their disposal leaned on their other experiences, citing ideas like "diffusion of responsibility" in large groups of people and suggesting the team define checklists and other procedures to keep the team focused. Novices who had the tool used it to help them categorize this finding and identify useful remediation measures. The novices were used to encountering problems they had no experience with and used all of the tools at their disposal to solve them, while the expert in this situation seemed to be derailed when he encountered a problem he was unfamiliar with. This finding is in contrast to some expert-novice studies [Doukakis, 2018], which have studied the behavior of these groups in an academic setting. Our study may yield different results because we were conducting this interview in an industry setting. Further study into how the study is framed may be beneficial to better frame the expert-novice behavior differences.

These two experts who used the tool while answering questions on findings from NASA failures were not interested in using the tool to go through each individual finding, and rather wanted to explore the information in the tool and compare it to what they knew already about the aerospace accidents. Despite having the tool available, they still relied on their expertise to provide recommendations on the problems. Björklund [2013] describes how "expert responses are centered around recognizing or constructing adequate responses based on experience and critical features of the problem, rather than exhaustively analyzing options to produce normatively optimal responses". The experts wanted to go through the tool in their spare time rather than when faced with a problem with the interviewer watching them. This is in contrast to the graduate students, who may be more used to constantly being observed and evaluated.

The novices also primarily focused on the aerospace accidents in the tool. A small subset of them explored accidents from other industries, but most wanted to sort the network by aerospace accidents right away. This could also be because they were answering questions on findings from aerospace failures, and most were unfamiliar with current theories in accident causation; that accidents share causes, regardless of industry.

4.4.3 Recommendation Rating Method

For each recommendation each participant made, we also asked them to rate on a scale of 1 to 10 how effective their recommendation would be at solving the problem the finding identified. We also need a more objective method of comparing each recommendation to determine whether our tool helped to improve the recommendation quality. Thus, we developed four measures of recommendation quality: specificity, scope, ease of implementation, and impact if successful. We rated each recommendation on a scale of 1 to 3. In all four measures, a higher score means the recommendation better encapsulates this quality, but note that a recommendation does not have to score highly in all four qualities to be effective, successful, or useful to companies experiencing problems. Table 4.8 contains descriptions of these four qualities, as well as examples from the past failures we studied describing low and high scores for each quality.

Table 4.8: Recommendation measure descriptions

Measure	**Description**	**Example of low score**	**Example of high score**
Specificity	Whether the recommendation gives explicit steps to take.	**Fukushima Nuclear Disaster** "Existing laws should be consolidated and rewritten in order to meet global standards of safety, public health and welfare." [Kurokawa et al., 2012, p. 23]	**Alaska Airlines Flight 261 Crash** "Establish the jackscrew assembly lubrication procedure as a required inspection item that must have an inspector's signoff before the task can be considered complete." [NTSB, 2000, p. 181]
Scope	How broad the recommendation is; can it be applied throughout the organization?	**Exxon Valdez Oil Spill** "Require that two licensed watch officers be present to conn and navigate vessels in Prince William Sound." [NTSB, 1989b, p. 171]	**Aloha Airlines Flight 243 Crash** "Revise the National Aviation Safety Inspection Program objectives to require that inspectors evaluate not only the paperwork trail, but also the actual condition of the fleet airplanes undergoing maintenance and on the operational ramp." [NTSB, 1989a, p. 75]
Ease of Implementation	How much time, effort, or resources are required to carry out the recommendation.	**Fukushima Nuclear Disaster** Establish a “new regulatory organization [that is] independent, transparent, professional, consolidated, and proactive." [Kurokawa et al. 2012, p. 23]	**Upper Big Branch Mine Collapse** "Digital photographs from recent inspections and other appropriate visual aids should be used to demonstrate to miners, managers and inspectors acceptable and non-acceptable mining equipment and conditions." [McAteer et al. 2010, p. 111]

Table 4.8 continued

Impact if Successful	How effective the recommendation would be at fixing the underlying problem.	**Baia Mare Gold Mine Cyanide Spill** "A risk assessment study should be carried out of the entire system of remining the old tailings." [United Nations Environment Programme and the Office for the Co-ordination of Humanitarian Affairs, 2000, p. 47]	**Texas City Refinery Explosion** "Configure control board displays to clearly indicate material balance for distillation towers." [CSB, 2007, p. 215]

We need to determine how objective our rating scheme is, similarly to how we analyzed our grading scheme for the student survey responses in Section 3.3.2. However, since we collected a smaller number of responses to these interviews than the survey, we had the additional graders rate all of the recommendations we collected instead of simply a subset of these recommendations. We asked four individuals to rate the recommendations; two people split the Helios recommendations and two people split the ISS recommendations. To calculate the inter-rater agreement, we calculated the Kappa Index, which is one of the original and most commonly used inter-rater agreement indices and corrects the agreement value by subtracting what the graders could assign by chance [Shweta et al., 2015][19]. Equation (7) describes the method we used to calculate the Kappa (K) index.

$$K = \frac{Observed\ agreement - Chance\ agreement}{1 - Chance\ agreement} \tag{7}$$

For each recommendation rating, the chance agreement is 11% (the chance of assigning a 1, 2, or 3 to a recommendation is $\frac{1}{3}$, and since the grading was done independently, we multiply these together: $\frac{1}{3} * \frac{1}{3} = \frac{1}{9}$). Table 4.9 contains the observed agreement value between the two graders and the kappa value we calculated for each category.

[19] We use a different inter-rater analysis method here than in Section 3.3.2 because here we have the additional graders rate all of the responses, as opposed to having them rate a subset of the responses. We were able to do this because the number of responses to the usefulness interviews was far fewer than the number of responses to the survey.

Table 4.9: Inter-rater agreement results

Category		**Observed Agreement**	***K***
Overall		0.49	0.42
By measure	Specificity	0.47	0.41
	Scope	0.43	0.36
	Ease	0.52	0.46
	Impact	0.52	0.46
By individual	Helios Rater 1	0.41	0.34
	Helios Rater 2	0.48	0.41
	ISS Rater 1	0.52	0.46
	ISS Rater 2	0.54	0.48

Shweta et al. [2015] state that a Kappa value of 0.40 or less is indicative of poor agreement, and values between 0.40 and 0.75 represent fair to good agreement. Thus, the overall agreement is fair, while some of the categories in Table 4.9 have poor agreement. The measure that requires the most work is scope, which has a Kappa value of 0.36. This result indicates that our rating scheme requires work to become more objective, and we outline a plan for improving it in Section 5.

4.4.4 Recommendation Analysis and Comparison

The recommendation rating system we developed helps to capture qualities that different recommendations have that help make them effective at fixing problems in systems engineering failures. The recommendations we discussed in Table 4.8 were all made by experts in accident causation, so how do the engineers who wrote the report on their own failures, as well as the experts and novices in systems engineers we interviewed compare? To answer this question, we first need to answer the following questions:

1. How did the expert and novice recommendation scores compare?
2. Did using the tool improve the quality of individual respondents' recommendations from one failure to another?
3. How do the expert, novice, and NASA recommendations compare qualitatively?

How did the expert and novice recommendation scores compare? Figure 4.8 shows a scatterplot of the expert and novice recommendation ratings for each of the four recommendation measures.

It is difficult to directly compare this data because we have different numbers of responses for each group (4 experts and 17 novices).

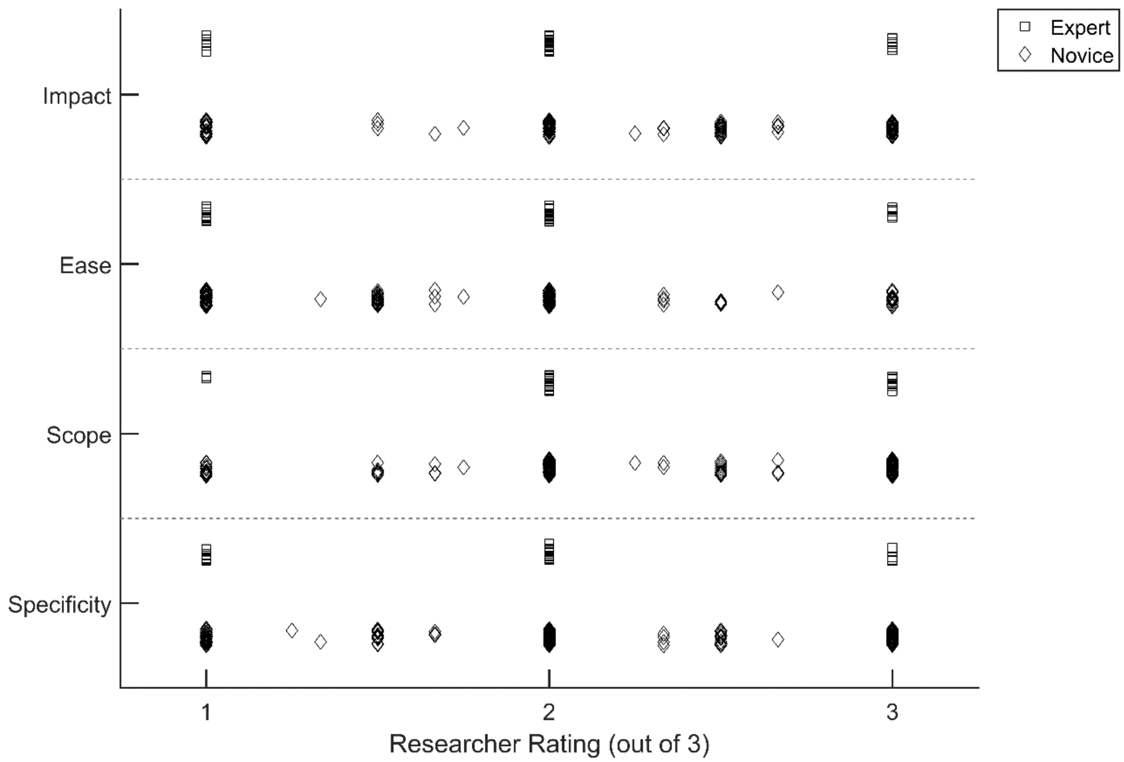


Figure 4.8: Expert and novice recommendation ratings

Figure 4.8 shows that both the novices and the experts had ranges of scores over all four recommendation measures. However, some novices gave multiple recommendations on a single finding, and we averaged these scores to give some value between 1, 2, or 3. The experts did not give multiple recommendations on a single finding at all, which reflects our discussion in Section 4.4.2, where we stated that the experts had more trouble than the novices distinguishing between subtleties in the findings and gave similar recommendations for multiple findings. The experts had a single idea for each problem that they recommended and thus received a single score for each one.

Figure 4.9 shows a scatterplot of the ratings the experts and novices gave their own recommendations out of ten. This plot shows that experts generally rated their recommendations slightly higher than novices, which may indicate that confidence in the effectiveness of recommendations an engineer gives is a function of experience.

Figure 4.9: Expert and novice self-ratings on recommendations

Did using the tool improve the quality of individual respondents' recommendations from one failure to the next? Table 4.10 shows the changes in ratings for novice responses from one failure to the next, for all four scenarios as described in Figure 4.6. Each box indicates a difference in averages between the scores novices received for recommendations they gave on the first failure they did and the second failure they did. For example, novices who answered questions on the Helios failure first received an average specificity score of 2.12. Novices who answered question on the ISS water intrusion failure second and the difference between these two averages is –0.06, which rounds to approximately –0.10. In Table 4.10, a red box indicates a negative difference (the score on the **first** failure is higher) and a green box indicates a positive difference (the score on the **second** failure is higher).

Table 4.10: Novice recommendation rating improvement from one failure to another

Study Design	First Failure (all without tool)	Second Failure	Self-Rating	Specificity	Scope	Ease	Impact
CD	Helios	ISS water intrusion (no tool)	-0.1	-0.1	-0.1	-0.1	-0.1
CB	Helios	ISS water intrusion (with tool)	0.5	-0.3	0.2	-0.6	0.1
DC	ISS water intrusion	Helios (no tool)	0.4	0.1	0.2	-0.2	0.1
DA	ISS water intrusion	Helios (with tool)	0.2	-0.3	0.0	0.1	0.0

As we discussed in Section 4.4.2, the novices in general were more familiar with the Helios failure than the ISS water intrusion failure. This is because most of the graduate students' work was in aviation research—only two of the students majored in astrodynamics and had specific space applications for their research. This likely explains the result for study design CD: since the novices were more familiar with the Helios failure, the recommendations they gave for this failure scored higher in each quality and they scored their own recommendations higher when they did not have the tool helping them with the ISS water intrusion failure.

When the novices had the tool helping them with the ISS water intrusion failure for study design CB, their recommendation scores improved slightly for *scope*, *impact*, and their self-ratings. However, their scores decreased for *specificity* and *ease of implementation*.

For study design DC, when the novices answered questions on the failure they were more familiar with second, their scores improved as expected for most of the recommendation qualities, except for *ease of implementation*.

When the novices had the tool helping them with the Helios failure for study design DA, their recommendation scores improved slightly for *ease of implementation* and their self-ratings. The novices' scores decreased for *specificity*, as they did for study design CB. The novices' scores had minimal differences for *scope* and *impact*, which is in contrast to the results for study design CB. This result indicates that users needing help with an unfamiliar problem on a project could use the tool to help their remediation measures have larger *scope* and *impact*. However, they should review their remediation measures to ensure they do not score low in *ease of implementation*. When users need help with a more familiar problem on a project, the tool may help their score in *ease of implementation*. In either case, users should ensure their recommendations do not score low in *specificity*. Additionally, the tool helped interviewees' self-ratings and may have boosted their confidence in their own recommendations.

For the Helios failure, how do the expert, novice, and NASA recommendations compare qualitatively? Table 4.11 contains one of the five findings we used in the interviews from the Helios report, the recommendations the NASA employees who wrote the report, and the recommendations from the experts and novices we interviewed gave for the finding.

Table 4.11: Recommendation examples for one Helios finding

Finding from Helios report						
Finding from Helios report: The Helios crew did not recognize that the aircraft was becoming unstable during flight. This was partially because there were no clear roles and responsibilities if instability was recognized: people were present, but were not explicitly told to give input on the flight. The crew was overconfident that the flight would succeed because of past flight success [paraphrased from NASA, 2004, p. 85].						
Description	**Response**	**Self-rating**	**Spec-ificity**	**Scope**	**Ease**	**Impact**
Excerpt from recommendation section of NASA report	"Further refine the roles and responsibilities of the crewmembers to improve overall team response to unexpected and anticipated emergency conditions. Refine emergency recognition criteria to improve team emergency response. Perform simulations to develop recognition criteria that identify the vehicle's response prior to and during instabilities. Improve the fidelity of aircrew simulations to mitigate the risks associated with takeoff and landing." [NASA 2004, pp. 90-91]	N/A	3	2	3	3
Expert 04 (did not use tool)	Circle back all the way to con-ops (concept of operations): you need to establish the roles and responsibilities of the crew then, and use that information to model the user environment. Do you have enough automation and/or crew?	4.5/10	1	3	2	2
Novice 05 (did not use tool)	Change the team cooperation architecture. Right now they have more of a hierarchy with a group leader, team leader, and manager. Everyone is very focused on their own job. A flattened cooperation pattern or architecture may be more helpful. They can take turns being the "challenger", challenging the decisions the team makes and asking questions. The other people can also evaluate the performance of the challenger.	7/10	3	3	2	2
Novice 08 (used tool)	First: "Give supervisor more capacity for oversight" there should be a defined clear role for what the supervisor's role is. Second: "Make instructions more clear" as in who should do what on the team. Who has the role of correcting problems on the flight? Third: There was no way to inform the control room that there was a problem if someone recognized it. They needed better communication channels.	9/10	1.7[20]	2.3	3	1.7

[20] Some of the interviewees gave multiple recommendations for a single finding. In these cases we rated each recommendation separately and averaged their scores in each category.

The responses in Table 4.11 all touch upon similar ideas: that the team's roles and responsibilities needed to be further refined in advance. Interestingly, both the expert and the NASA recommendations suggested modeling or simulating the environment in some way, which was a concept that none of the 17 novices we interviewed discussed. Two novices recommended improving the training for emergency situations, and many of the novices recommended that the team increase automation in some way. The experts combined these ideas in their recommendations. Alternatively, many of the novices discussed ideas relating to how teams behave (Novice 05 mentioned "team cooperation architecture), which none of the experts discussed.

For the ISS water intrusion failure, how do the expert, novice, and NASA recommendations compare qualitatively? Table 4.12 contains one of the five findings we used in the interviews from the ISS Water Intrusion report, the recommendations the NASA employees who wrote the report, and the recommendations from the experts and novices we interviewed gave for the finding.

Table 4.12: Recommendation examples for one ISS Water Intrusion finding

Finding from ISS water intrusion report: The spacesuit had filled with water on the previous EVA, but there was not a lot of time before the next EVA. So, the ground team decided to perform the next EVA without finding out what had happened. They wanted to avoid initiating a lengthy formal risk assessment process, which in may have found the real source of the water and avoided the dangerous scenario [paraphrased from NASA, 2013, p. 84].

Description	**Response**	**Self-rating**	**Spec-ificity**	**Scope**	**Ease**	**Impact**
Excerpt from recommendation section of NASA report	"The ISS Program must reiterate to all team members that, if they feel that crew time is needed to support their system, a request and associated rationale must be elevated to the ISS Program for an appropriate decision." [NASA 2013, p. 145]	N/A	2	2	2	1
Expert 01[21] (did not use tool)	They should have used a board of experts, held a meeting, and then decided what to do from there. First they should have followed the process: identify the risk, decide how bad the risk is, mitigate the risk (or end the testing).	10/10	2	3	2	2

[21] The responses are from different interviewees than from those in Table 4.11.

<table>
<tr><td colspan="7">Finding from ISS water intrusion report: The spacesuit had filled with water on the previous EVA, but there was not a lot of time before the next EVA. So, the ground team decided to perform the next EVA without finding out what had happened. They wanted to avoid initiating a lengthy formal risk assessment process, which in may have found the real source of the water and avoided the dangerous scenario [paraphrased from NASA, 2013, p. 84].</td></tr>
<tr><th>Description</th><th>Response</th><th>Self-rating</th><th>Spec-ificity</th><th>Scope</th><th>Ease</th><th>Impact</th></tr>
<tr><td>Novice 13
(did not use tool)</td><td>They should consider all the possibilities for all potential sources for water in the spacesuit, then eliminate each one by investigation. Time spent should not be a factor in whether they decide to do this because it is important.</td><td>9/10</td><td>2</td><td>2</td><td>1</td><td>1</td></tr>
<tr><td>Novice 10
(used tool)</td><td>If the regulations were more strict, they wouldn’t have been able to go forward with the mission without doing the risk assessment process.
One of the reasons they didn’t do the risk assessment was because it was lengthy and difficult to do, as well as time consuming, so if they had a better way of doing this it would be more effective. Perhaps in a systematic way to reassess the design of the helmet, they would have done it. They had to figure out a method, form a team, and do the risk assessment, which is tedious and time consuming.</td><td>5/10</td><td>2</td><td>3</td><td>1</td><td>2</td></tr>
</table>

Similar to the Helios failure, the responses in Table 4.12 touch upon similar ideas: that the team should have investigated the problem to find the real source of the water and eliminate it. Again, the expert and the NASA recommendations were similar in that both suggested the team follow the process to eliminate the problem. Many of the novices (including all of the novices who used the tool) criticized the risk process itself and suggested that it be improved to encourage the team to follow it in the future. The experts may not have suggested this remediation measure because they understand better than novices how difficult it can be to change processes at a company.

4.5 Summary

In this section we developed an interactive, web-based expertise aid based on our study of past systems engineering failures that presents users with causes of failures, as well as recommendations from experts in failure causation. Systems engineers experiencing problems on their projects could use this tool to see problems other projects experienced, and what accident investigators recommended these projects do to solve their problems. We also studied whether the tool was useful for people experiencing failures by interviewing systems engineers of different expertise levels as they used the tool to provide recommendations on findings from NASA failures. We found that overall the tool was the most useful for people more unfamiliar with a failure. Most of the people we interviewed were more familiar with aeronautical concepts over astronautical concepts, so they relied less on the tool to make recommendations on an aircraft failure and relied more on the tool to make recommendations on a space applications failure. Not only did the tool help these people make better recommendations on our more objective scale, but also the tool made the interviewees more confident in their recommendations.

For additional feedback we received on the tool that we did not discuss in this section, refer to Appendix G.

CHAPTER 5. CONCLUSIONS AND FUTURE WORK

In Section 2 we identified a set of 30 accidents and 33 project failures, spanning a wide range of industries. Next, we modified Leveson's STAMP model and used it to methodically extract and analyze their causes. We identified 23 different accident and project failure causes and 16 different recommended remedial actions across the 63 cases. We presented our findings as a cause-recommendation network, which shows how the failure causes are linked to each other and to the recommendations, and also provides over 900 specific examples of how these causes manifested in failures, and over 600 specific examples of the associated recommended remedial actions. We interviewed experienced systems engineers at large-scale aerospace companies to determine how the problems they encountered compared to the problems we identified that led to systems engineering failures.

The limitations of this study are such: first with identifying project failures to study. As Judgev & Müller [2005] state in their paper on understanding project success: "Trying to pin down what success means in the project context is akin to gaining consensus from a group of people on the definition of 'good art'." Not only is project success difficult to define, but project failure is also not one-minus the definition of project success. Readers may disagree with the way in which we defined project failures (e.g. we classified unmanned space mission failures as project failures, but we classified the Space Shuttle disasters in which the crews were killed as accidents), but this distinction has no material effect on our results and our results are potentially useful for any project experiencing problems, no matter the distinction. Second, studying a set of previously-reported project failures and accidents is inherently subject to bias from the investigators. These biases are inherent to any approach based on studying investigation reports. We discuss these potential biases at length in Sorenson [2015]. Third, the extraction and coding process is subject to bias by the coders. Different coders may identify more or fewer causes or recommendations in a given report, and different coders may assign a given finding or recommendation to different codes. Since we provide in the network both the original sources and the paraphrased "stories" behind each instance of each code, the impact of the code creation and allocation process is minimal.

In general, it is difficult to make concrete conclusions on the Company A data because (1) we do not know how many systems engineers participated in the single paper survey response we received, and (2) we were only given limited data on five failures at the company. This may explain the discrepancies between the Company A data analysis, our analysis of past failures, and the results of the paper survey. For example, both our study of past failures and the respondents' paper survey results identified problems with risk management, but we only identified this cause in a single failure described by Company A (note that when we used the paper survey coding scheme, we identified "ineffective risk management procedures" in 3 failure narratives; our coding scheme allowed for more specific codes to be assigned to each finding). Overall, it makes sense that Company A provided supplementary data on failures they experienced because many of the ideas represented in that data were difficult to translate to the paper survey. We struggled to directly compare the paper survey coding scheme to the coding scheme we developed from studying past systems engineering failures. Perhaps if we developed a new survey more closely aligned to these causes other companies will also find it easier to compare their project failure performance.

Through this study we discovered that different companies are willing to share different information in different formats. Representatives at Company B found the general language we used in our original survey to be too restrictive to their responses and wanted the opportunity to respond to questions in a more freeform format like an interview. Representatives at Company A were concerned about data privacy and thus gave us information they pre-approved, rather than having us interview their employees. To approach other companies in the future to continue investigating systems engineering, we should be prepared to tailor our survey methods to each company's needs.

In this section, we focused on analyzing and discussing the causes of systems engineering failures. In future work, we will expand the network by incorporating other aspects from our analysis, for instance (1) The actors involved in each cause, (2) The types of objects involved in the causes and the difference between project failures and accidents (e.g. what types of testing was involved), or (3) When in the design cycle the cause occurs. Companies experiencing problems during project development may use the cause-recommendation network as a guide to analyze any issues they have found, identify other potential related issues, and then use the recommendation codes to reduce the likelihood of failure.

Part of our work in this section involved developing a specialized coding scheme to compare the causes of systems engineering related accidents and project failures. There are also other coding schemes, both more general and more specific, such as the HFACS accident causation hierarchy. In our future work we plan on mapping our coding scheme to other methods to analyze the differences in the coding schemes and determine whether different patterns emerge.

Currently, adding stories to the network is easy, but extracting and coding them requires more effort. Past research has been conducted on using machine learning to analyze accident reports [Abedin et al., 2010; Ghaoui et al., 2013; Robinson et al., 2015] so we may be able to use a similar method to teach a machine our coding scheme and easily add failures to our cause-recommendation network.

As stated in Section 2.5, our cause-recommendation network is similar in construction and representation to a Bayesian network. Future work for this research may be using tools like BayesiaLab to analyze our data and represent it in a Bayesian network format [Conrady & Jouffe, 2015], which may allow us to also incorporates ideas like the probability of a failure occurring depending on how many causes are found and in what combination they occur.

In Section 3 we described the responses from the systems engineers from Company B we interviewed gave on what discrepancies they have identified in systems engineering. We then developed 8 survey questions based on decision errors in systems engineering, how we distributed this survey, and the responses we received. We conducted a statistical analysis on the data using the linear regression and ordinal logistic regression models and we found what variables were statistically significant for each survey question and interpreted what each variable meant based on its intercept value. We built classification and regression tree diagrams for each survey question to visually compare how each variable branched in the data. We then described the range of systems engineering content in survey responses we received using illustrative examples.

Does survey performance relate at all to systems engineering course performance? Our results may benefit from further data collection, although modified by what we have learned so far. For example, one weakness in our data is that the two populations of students who took the survey, undergraduate and graduate students, require different analysis methods because the graduate student population do not have as much systems engineering-related course data. We are investigating other avenues of analyzing and comparing these differences, such as by collecting other class data, or asking the graduate students to self-identify systems courses they have taken in their undergraduate studies or design projects they have worked on. We would also like to determine whether we can use this survey (or a version of it) to measure how systems engineering skills change over time. As we have collected many semesters' worth of responses to the survey, some students have responded to the survey multiple times. However, most of these responses were incomplete and it was difficult to measure how the student improved from one response to the other. We suspect this is because the students recognized the survey questions and did not feel they should re-do their responses. Perhaps we need to build a bigger bank of questions that are based on similar topics but are not easily recognized from one semester to another so answering the survey multiple times does not seem so repetitive to the respondents.

Measuring survey performance in relation to systems engineering course performance was the first step in our process to eventually using this survey across engineering disciplines and even outside of education applications. We would also like to continue working on our survey to determine whether we can use it to measure systems engineering ability, such as for industry representatives who may have solely received their systems engineering training on-the-job. While we compared the range of responses that demonstrated systems engineering qualities in Section 3.4.3, further comparison between responses for these proposed applications may benefit from a more detailed analysis method beyond our quality grading scheme. Latent semantic analysis is a fully automatic statistical approach to comparing written responses and we may be able to use this technique in the future to more accurately analyze these responses [Dumais, 2004].

In Section 4 we developed an interactive, web-based expertise aid based on our study of past systems engineering failures that presents users with causes of failures, as well as recommendations from experts in failure causation. We described potential applications of the tool, as well as some of its functions. We also studied whether the tool was useful for people experiencing failures by interviewing systems engineers of different expertise levels as they used the tool to provide recommendations on findings from NASA failures.

We found that our recommendation rating scheme of assigning a value of 1, 2, or 3 to four measures (specificity, scope, ease of implementation, and impact if successful) requires work to be more objective. In particular, raters disagreed the most on "scope". We could improve this rating scheme by changing it to a quality-based format like we did for the survey grading scheme in Section 3.3.1. For scope, raters could tick a box for each of the four qualities: (1) recommendation applies to part/specific problem, (2) recommendation applies to project, (3) recommendation applies to company as a whole, and (4) recommendation applies to the industry. These qualities get gradually broader and broader in scope and help the rater identify exactly how broad the recommendation is. For ease of implementation, raters could tick a box for each of these four qualities: (1) recommendation could be carried out in a short period of time, (2) recommendation could be carried out with no additional personnel, (3) recommendation could be carried out with minimal additional funding, and (4) recommendation could be carried out with minimal development efforts (e.g. to build tools or research concepts). This would also help identify what recommendations may be the most useful for industry applications; an organization like NASA may have trouble hiring more personnel and getting additional funding, but have the resources available to develop and research new methods and not have tight time constraints on their projects.

A quantitative analysis of the differences in recommendation scores indicated that the interviewees' scores changed depending on whether they had access to the tool, and their scores may improve differently depending on whether they are familiar with the failure they encountered. However, most of the values in Table 4.10 are not high in magnitude, so it is difficult to make concrete conclusions about objective improvements in scores. Further study using failures from industries that the interviewees are more unfamiliar with may be useful, such as interviewing aerospace engineers using findings from oil and gas failures and vice-versa. A qualitative analysis of the differences in topics from each recommendation revealed that most touched upon similar ideas, and that the experts and novices each identified novel, ideas in different ways. In general, the tool helped the interviewees the most when they were unfamiliar with a topic.

One aspect of using the two NASA failures was that we noticed that, in general, the Helios report gave recommendations that scored higher in all four categories than the ISS water intrusion report recommendations. For the examples shown in Table 4.11, the interviewees gave recommendations that scored higher than the ones given in the ISS water intrusion report, but had recommendations that did not score as high as the ones given in the Helios report. In future work we could incorporate this recommendation rating system into the tool to help systems engineers find help in areas in which they are most weak. For example, if the engineers know that their remediation measures have not had much impact in the past they could sort recommendations in the tool by high scores in “impact if successful” to see what experts have said on that topic.

We received a lot of feedback on additional functions the tool could have, as well as many ideas for potential applications of the tool from the people we interviewed. We plan on investigating and implementing some of this feedback into the tool in future work.

APPENDIX A. COMPANY SURVEYS

Survey on Failures in Engineered Systems

Introduction

Dear Participant,

As part of a research study, we are interested in understanding how systems engineering occurs in practice in industry.

This study is in collaboration with the Purdue School of Aeronautics and Astronautics (PI: Dr. Karen Marais, kmarais@purdue.edu). Your participation is much appreciated.

This study has been approved by the Purdue University Institutional Review Board (irb@purdue.edu).

The survey should take you approximately 30 minutes to complete. Please do not include any confidential company information in your response. You may consult with NAME OF CONTACT AT COMPANY if you have any questions about your responses. **Please note that your responses are not confidential and must be reviewed for public release in compliance with your organization's policies**.

Your organization will be referred to as "aerospace company X", where X is a random alphanumeric character. The PI will not have access to any identifying information or individual participants' data.

Your participation is entirely voluntary. You may respond to as many questions as you like. If at any time you do not wish to participate you may dispose of this document as you wish. When you have completed the survey, please return it to NAME OF CONTACT AT COMPANY for clearance for public release. Once your survey has been cleared you are welcome to retain a copy for your records. NAME OF CONTACT AT COMPANY will return the compiled responses to the PI.

Thank you in advance for your time and participation! If you have any questions about this study feel free to contact NAME OF CONTACT AT COMPANY or the PI at kmarais@purdue.edu.

Consent

I have had the opportunity to read this consent form and have the research study explained. I have had the opportunity to ask questions about the research study, and my questions have been answered. I am prepared to participate in the research study described above.

Yes: _______

No: _______

Systems Engineering Failures

We are interested in the following types of engineering failures:

- Accident/Incident
    - For example, on 23 September 1999, communication with the Mars Climate Orbiter was lost as the spacecraft went into orbital insertion. The insertion failed because of a units mismatch between NASA and Lockheed. The spacecraft encountered the Martian at an improperly low altitude, causing it to incorrectly enter the upper atmosphere and disintegrate.
- Quality Concerns
    - For example, Toyota issued a recall in January 2010 due to possible mechanical sticking of the accelerator pedal causing unintended acceleration.
- Cancellation
    - For example, the X-33 was cancelled in 2001 after a long series of problems with flight stability and excess weight.
- Performance gap
    - For example, Iridium filed for bankruptcy in 1999 after it failed to garner enough subscribers. This failure was due in part to poor phone coverage. Because the technology depended on line-of-sight between phone antennas and the orbiting satellites, the phones did not work inside moving cars, inside buildings, and in many urban areas. Iridium was subsequently reborn, but at a much smaller scale than originally envisioned.
- Delay and cost overrun
    - The Boston Big Dig was originally scheduled to be completed in 1998 at an estimated cost of $2.8 billion (1982 dollars). The project was plagued with technical, scheduling, cost, and even criminal problems. It was eventually completed in in December 2007, at about 190% of the originally planned budget (over $14.6 billion in 2006 dollars).

Professional Background

1. Please identify your position within the organization. You can be as general or specific as you like.
2. How long have you been practicing as an engineer?
3. How long have you been practicing as an engineer with systems level visibility or accountability?

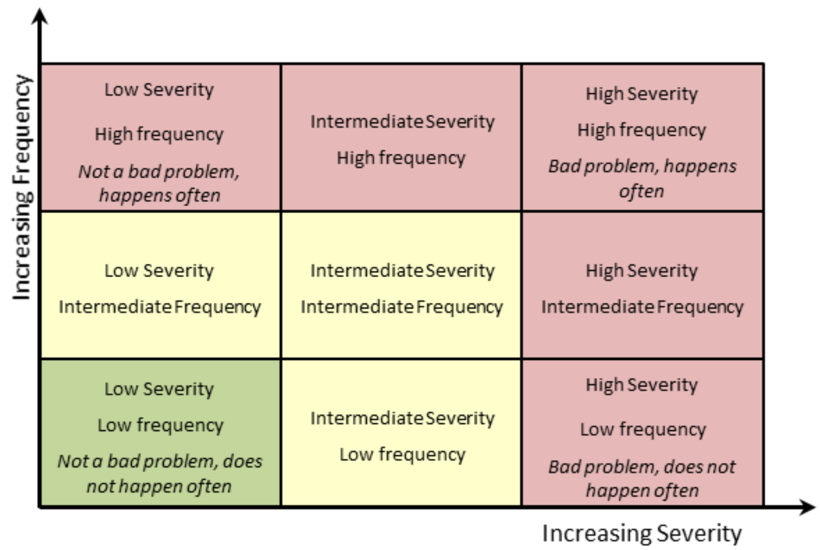
What proportion of projects, both that you have worked on, and that you have observed either at your own organization or elsewhere, have suffered from these problems? Note that a given project may suffer more than one type of failure, so the percentages do not need to add to 100%.

Type of Event	Percentage of Projects suffering from Event	Prefer not to answer
Accident/Incident		
Quality Concerns		
Cancellation		
Performance gap		
Delay		
Cost overrun		
No significant failures		

Thinking about these events, how often have they been related to systems engineering?

Type of Event	Percentage of Events related to Systems Engineering	Prefer not to answer
Accident/Incident		
Quality Concerns		
Cancellation		
Performance gap		
Delay		
Cost overrun		
No significant failures		

The next six questions are concerned with how systems engineering can contribute to different types of failures. For each of the issues under the systems engineering failures, rate its importance on the following scale.



For example, consider "not using lessons learned." On the next page are examples about this issue and how it generally affects each systems engineering failure at a fictitious company, Widgets Inc. Your company may be differently impacted by issues in each failure category.

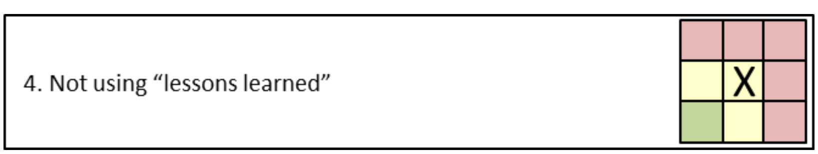
Previous research has shown these kinds of problems are often the cause of Systems Engineering failures.

If you are confused about what a category may mean, please refer to the glossary at the end of this document.

Examples

Accidents and Incidents

At Widgets Inc., not using lessons learned has an intermediate impact upon the occurrence of accidents and incidents. These "lessons learned" could be from previous accidents and incidents, e.g. if people were slipping on mopped floors and no signage was provided after these incidents occurred.



Quality Concerns

"Not using lessons learned" does not affect quality concerns very often at Widgets Inc. but when it does, it has a moderate impact. Parts being made are sometimes not at a desirable quality level, but the underlying quality problem is not addressed and applied to future parts being made.

Cancellations

I have no opinion on this issue or I do not wish to answer.

Performance Gaps

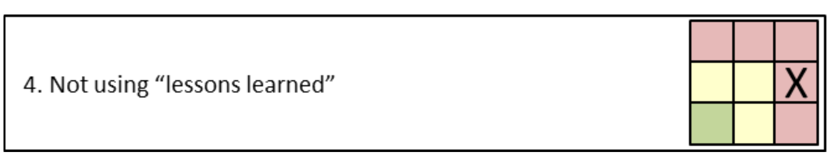
At Widgets Inc., performance gaps due to “not using lessons learned” have a moderate impact with little frequency. The engineers at Widgets Inc. do not use lessons learned from previous projects and the performance of the current project suffers.

Delays

Delays at Widgets Inc. are affected somewhat often by “not using lessons learned” but its impact is minimal at Widgets Inc. The workers sometimes work overtime to make up for delays due to this issue.

Cost Overruns

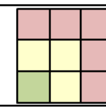
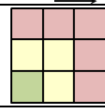
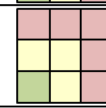
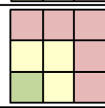
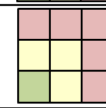
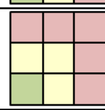
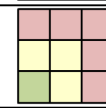
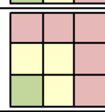
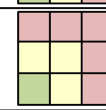
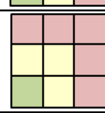
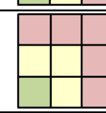
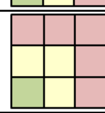
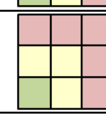
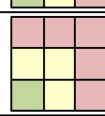
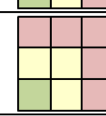
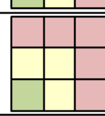
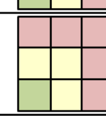
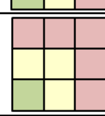
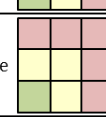
At Widgets Inc., “not using ‘lessons learned’” has a very large impact upon cost overruns, possibly because employees are putting in a lot of overtime to correct mistakes that could have been avoided. Additionally, parts have to be redone or scrapped because mistakes are being made and aren’t corrected.



In the projects you have been involved in at your organization, how do each of the issues below apply to **accidents and incidents**?

A1. Weakness in otherwise good processes (Increasing Frequency / Increasing Severity)	A11. Inadequate knowledge transfer
A2. Loss of company knowledge due to employee retirement	A12. Lack of employee loyalty
A3. Inability to hire enough systems engineers	A13. Issues with reward and incentive structures
A4. Not using “lessons learned”	A14. Ineffective risk management procedures
A5. Process inadequate for complex systems	A15. Ineffective team building
A6. Inadequate planning in the early stages	A16. Acquisition management failures
A7. Problems with staff, training, or expertise	A17. Ineffective government/industry teaming
A8. Lack of rigor	A18. Real time efficiency decisions that led to later problems
A9. Ineffective resource allocation	A19. Dysfunctional feedback across the system lifecycle
A10. Inadequate requirements engineering	A20. Using process rather than thinking and being accountable

In the projects you have been involved in at your organization, how do each of the issues below apply to **quality concerns**?

B1. Weakness in otherwise good processes		B11. Inadequate knowledge transfer	
B2. Loss of company knowledge due to employee retirement		B12. Lack of employee loyalty	
B3. Inability to hire enough systems engineers		B13. Issues with reward and incentive structures	
B4. Not using "lessons learned"		B14. Ineffective risk management procedures	
B5. Process inadequate for complex systems		B15. Ineffective team building	
B6. Inadequate planning in the early stages		B16. Acquisition management failures	
B7. Problems with staff, training, or expertise		B17. Ineffective government/industry teaming	
B8. Lack of rigor		B18. Real time efficiency decisions that led to later problems	
B9. Ineffective resource allocation		B19. Dysfunctional feedback across the system lifecycle	
B10. Inadequate requirements engineering		B20. Using process rather than thinking and being accountable	

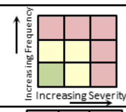
In the projects you have been involved in at your organization, how do each of the issues below apply to **cancellations**?

C1. Weakness in otherwise good processes (Increasing Frequency / Increasing Severity)	C11. Inadequate knowledge transfer
C2. Loss of company knowledge due to employee retirement	C12. Lack of employee loyalty
C3. Inability to hire enough systems engineers	C13. Issues with reward and incentive structures
C4. Not using "lessons learned"	C14. Ineffective risk management procedures
C5. Process inadequate for complex systems	C15. Ineffective team building
C6. Inadequate planning in the early stages	C16. Acquisition management failures
C7. Problems with staff, training, or expertise	C17. Ineffective government/industry teaming
C8. Lack of rigor	C18. Real time efficiency decisions that led to later problems
C9. Ineffective resource allocation	C19. Dysfunctional feedback across the system lifecycle
C10. Inadequate requirements engineering	C20. Using process rather than thinking and being accountable

In the projects you have been involved in at your organization, how do each of the issues below apply to **performance gaps**?

D1. Weakness in otherwise good processes (Increasing Frequency; Increasing Severity)	D11. Inadequate knowledge transfer
D2. Loss of company knowledge due to employee retirement	D12. Lack of employee loyalty
D3. Inability to hire enough systems engineers	D13. Issues with reward and incentive structures
D4. Not using "lessons learned"	D14. Ineffective risk management procedures
D5. Process inadequate for complex systems	D15. Ineffective team building
D6. Inadequate planning in the early stages	D16. Acquisition management failures
D7. Problems with staff, training, or expertise	D17. Ineffective government/industry teaming
D8. Lack of rigor	D18. Real time efficiency decisions that led to later problems
D9. Ineffective resource allocation	D19. Dysfunctional feedback across the system lifecycle
D10. Inadequate requirements engineering	D20. Using process rather than thinking and being accountable

In the projects you have been involved in at your organization, how do each of the issues below apply to **delays**?



E1. Weakness in otherwise good processes	E11. Inadequate knowledge transfer
E2. Loss of company knowledge due to employee retirement	E12. Lack of employee loyalty
E3. Inability to hire enough systems engineers	E13. Issues with reward and incentive structures
E4. Not using “lessons learned”	E14. Ineffective risk management procedures
E5. Process inadequate for complex systems	E15. Ineffective team building
E6. Inadequate planning in the early stages	E16. Acquisition management failures
E7. Problems with staff, training, or expertise	E17. Ineffective government/industry teaming
E8. Lack of rigor	E18. Real time efficiency decisions that led to later problems
E9. Ineffective resource allocation	E19. Dysfunctional feedback across the system lifecycle
E10. Inadequate requirements engineering	E20. Using process rather than thinking and being accountable

In the projects you have been involved in at your organization, how do each of the issues below apply to **cost overruns**?

F1. Weakness in otherwise good processes	F11. Inadequate knowledge transfer
F2. Loss of company knowledge due to employee retirement	F12. Lack of employee loyalty
F3. Inability to hire enough systems engineers	F13. Issues with reward and incentive structures
F4. Not using "lessons learned"	F14. Ineffective risk management procedures
F5. Process inadequate for complex systems	F15. Ineffective team building
F6. Inadequate planning in the early stages	F16. Acquisition management failures
F7. Problems with staff, training, or expertise	F17. Ineffective government/industry teaming
F8. Lack of rigor	F18. Real time efficiency decisions that led to later problems
F9. Ineffective resource allocation	F19. Dysfunctional feedback across the system lifecycle
F10. Inadequate requirements engineering	F20. Using process rather than thinking and being accountable

Systems Engineering Needs

In your opinion, what is the most important change that is needed in systems engineering?

You may answer as briefly or fully as you wish.

Glossary

1. Weakness in otherwise good processes
   A single weakness can devastate a manufacturing process, for example.
2. Loss of company knowledge due to employee retirement
   Employees are a wealth of project history and engineering knowledge, which can be lost when an employee retires.
3. Inability to hire enough systems engineers
   Systems engineers can be useful in managing complex projects and processes.
4. Not using "lessons learned"
   This issue can be a problem in large or far-reaching companies where projects or departments do not necessarily communicate between each other. If a solution is found to a common problem it may not be communicated to other places the same problem is present.
5. Process inadequate for complex systems
   A process will not be as effective for systems of different complexities.
6. Inadequate planning in the early stages
   This issue can be detrimental to schedule, for instance, if there is optimistic planning in the early stages.
7. Problems with staff, training, or expertise
   It is desirable for a workforce to be completely versed in the products or systems and have adequate training.
8. Lack of rigor
   A generally "lax" company can have trouble with a variety of issues, like enforcing safety protocols.
9. Ineffective resource allocation
   Resources must be allocated appropriately, especially in a company with multiple projects at varying degrees of completion.
10. Inadequate requirements engineering
    Requirements must be defined in order to know how to design and test a product, for example.
11. Inadequate knowledge transfer
    For example, new engineers often graduate from school with only the basic tools to do engineering work, and they must be trained by their coworkers to do useful work on a project.

12. Lack of employee loyalty
    Employees must care about the companies' overall success and have a desire to do good work.
13. Issues with reward and incentive structures
    Incentives can for example prioritize schedule over quality, which is often more difficult to quantify.
14. Ineffective risk management procedures
    Risks are present at every company, but they become dangerous if they are underestimated, not identified, or not addressed properly.
15. Ineffective team building
    Engineers generally work in teams.
16. Acquisition management failures
    Improper acquisition management can result in a product that does not meet the desired needs, is over-budget or behind schedule, or does not meet the test standards.
17. Ineffective government/industry teaming
    Often there are a lot of administrative obstacles a company must work through in order to effectively do work with the government.
18. Real time efficiency decisions that led to later problems
    Employees can cut corners to save time, but this could be detrimental to aspects like safety and performance.
19. Dysfunctional feedback across the system lifecycle
    It is important to know about issues like an incorrectly interpreted requirement as early as possible so that work does not have to be redone.
20. Using process rather than thinking and being accountable
    Work instructions, for example, can be very useful when assembling a product. However, they are not useful for new and complex products.

Semi-Structured Interview

SYSTEMS ENGINEERING QUESTIONS

KAREN MARAIS
ASSISTANT PROFESSOR, AEROSPACE ENGINEERING
kmarais@purdue.edu

PURDUE UNIVERSITY

DIANE ALOISIO
PH.D. STUDENT
dsorens@purdue.edu

Hello, my name is Diane Aloisio and I am affiliated with Purdue University. I am doing my master's degree with Dr. Marais. My work focuses on understanding why engineering projects sometimes fail. One part of our work involves interviewing industry members about their systems engineering practices.

Today I am going to be asking you a few open-ended questions about systems engineering at your company.

Let me begin by explaining your rights and responsibilities as an interviewee:

1. *You should already have a copy of this script; if not please give me your email and I will send it to you before we continue.*
2. *Please do not include any confidential company information in your response.*
3. *You may consult with NAME OF CONTACT AT COMPANY if you have any questions about your responses.*
4. *Please note that your responses are not confidential. I will be writing your responses down, and provide them to your company's university liaison for review for public release.*
5. *Your organization will be referred to as "aerospace company X", where X is a random alphanumeric character. You will be referred to by your job title. Where organizations have obviously recognizable job titles, we will replace them with generic equivalents.*
6. *Your participation is entirely voluntary. You may respond to as many questions as you like. If at any time you wish to end this interview, simply say so, or hang up.*

Thank you in advance for your time and participation! If you have any questions about this study feel free to contact NAME OF CONTACT AT COMPANY, the PI at kmarais@purdue.edu, or the Purdue board that reviews interview studies at irb@purdue.edu.

Let's begin with some basic information.

Q: What is your job title?

Q: How long have you worked as a Systems Engineer at your company?

Q: Have you worked in other roles at your company?

Q: Do you have experience with Systems Engineering in organizations other than at your current company?

In a few conversations I've had with your company's systems engineers I've noted that how systems engineers are categorized is important to keep track of all that goes on in systems engineering.

Q: Which area of Systems Engineering do you work in?

Let's talk about general systems engineering training at your company.

Some people working as systems engineers didn't necessarily go to school for systems engineering.

Q: What is your general academic background?

Q: What schooling or other training did you receive before becoming a Systems Engineer?

One of our goals on this project is to try and teach systems engineering better.

Q: Have you noticed any areas systems engineers struggle with the most when they are first hired?

Q: (If yes) Why do you think they struggle with these areas?

Q: If there are areas new employees are weak in, do you know of any training your company provides to try and mitigate this?

Q: (If yes) Do you think there are ways in which your company training for new systems engineers can improve?

Q: Do you look for certain traits when hiring systems engineers? (If manager)

Q: Have you noticed certain traits in some systems engineers you respect at your company?

Q: What experiences have you had when interfacing with other core functional engineering areas?

Q: Have you observed compliance to Systems Engineering processes in other core functional engineering areas at your company?

Let's talk now about some of the systems engineering tools you use at your company. When I say tool, I mean things like QFD, Functional Flow Block Diagrams, Systems "V", or Weighted Objectives Method.

Q: What are some of the general processes and tools you use?

At Purdue we teach students about many systems engineering models and tools. We have a general systems methods course that students take prior to their capstone senior design course. We are interested in how useful these courses are, and how we could improve them.

Q: Did you take any sort of general systems methods course in school?

Q: What systems engineering or project management programs or classes does your company encourage you to attend?

Q: Have you encountered any models or tools that you think would have been easier to use if you had learned about them in an academic setting?

Let's talk more about some tools or gadgets you use at your company.

Q: What is a tool you find particularly useful or interesting?

Q: What strengths does that tool have?

Q: What flaws does that tool have?

Q: Have you used any models or tools that you learned about in school?

Q: What strengths does that model or tool have?

Q: What flaws does that model or tool have?

Q: Was that model or tool modified for special use at your company?

Q: Are there any tools you learned how to use in school that you think your company should use?

Q: Were there any tools you learned about in school that you did not find useful?

Engineering practices change over time. For instance, before computers we used drafting tables to create engineering drawings.

Q: Have you noticed any changes in systems engineering practices over time at your company?

Q: How difficult is it to implement process changes, and do they last?

Q: What process changes have been implemented based on past projects?

Q: How are the design processes different at your company than at other companies? Do you think this makes projects more successful?

Let's talk about changes you've had to make to a design when you're in the middle of the design process.

Q: How do you accommodate these changes? How do you make sure you don't have to change designs in the future?

Q: What are your main priorities in your work, and how do they align with your company's priorities?

Some students find systems engineering a little dry.

Q: Do you have any suggestions to make systems engineering more interesting to students?

Ok, next I would like to move onto talking about occasions when things haven't gone as you wanted, such as a shortfall, turnback, or escape.

Q: Could you tell me about a shortfall that you encountered that you found particularly difficult or interesting?

Q: What factors contributed to this shortfall?

Q: How did you make sure this factor would not lead to shortfalls in other projects?

Q: How did you deal with shortfalls in this particular project? What lessons learned did you come up with?

Q: How did you decide on this particular corrective action?

Q: Do you think this remedy will be useful for escapes on other projects?

Q: Is there any phase in a product's lifecycle where you think your company encounters the most shortfalls?

Q: What kinds of product safety or flight safety programs do you have at your company, and do you think they are effective?

Q: How do you resolve conflicts with suppliers?

Q: How do you measure your Systems Engineering capabilities against the industry?

Q: What is your process for transferring lessons learned?

That was the last of the questions I have for you, are there any other points you would like to make before we end this interview?

Thank you for your time. If you have any questions or concerns about this interview, please feel free to contact NAME OF CONTACT AT COMPANY, the PI at kmarais@purdue.edu, or the Purdue board that reviews interview studies at irb@purdue.edu. You will be provided with a copy of our write-up of your responses.

APPENDIX B. COMPANY B RESPONSES TO SEMI-STRUCTURED INTERVIEW QUESTIONS

Responses Pertaining to Systems Engineering Failure Analysis

Table 5.1: Process changes that have been implemented on past projects

Session	Respondent A	Respondent B
1	Risk reviews are done differently, and now they elevate the reviews to the program manager in formal meetings. This company now makes risk reviews part of the staff meeting. Verification documentation tools are now used.	There is far more emphasis on risk management over the last six months and an accurate perception that engineers are too optimistic in their scheduling. All the scheduling is optimistic and the company does not deliver products on time because engineers do not build time into the schedule. In general, there is no schedule float and all tasks are optimistically scheduled. The respondent is now working on a project that has a realistic schedule and a “best case” schedule. Engineers on that project design to the realistic schedule and hope for the “best case” schedule.
2	Technology use has been updated. Several years ago there was a directive to do documentation in DOORS. That required a lot of momentum to shift from Word to DOORS (because they had long-lead items, and did not know how to use DOORS).	
3	The company had varying processes that it changed to make more aligned; engineers now have to get internal approval before asking any of their customers whether to change a requirement. The goal is to get aligned internally before talking to customers.	The company’s compliance form process has become more formal. In the past their compliance tracking process was more ad hoc, but now the company has a document trail that accompanies submitted documents as evidence that requirements are met. Specifically, the form has specific areas where compliance is shown.
4	20 years ago for a complex program the company had a requirements management tool they developed that mapped all the requirements to an analysis and had a complete checklist that showed that the design met every specification.	The change management system has been improved and fairly well-standardized.
5	She doesn’t know offhand	He doesn’t know offhand.

Table 5.2: Accommodating a design change

Session	Respondent A	Respondent B
1	Lessons learned: the international programs are less structured than the US government programs. The US programs have a customer “breathing down your neck”. International program customers are more “hands off”. Now the company has a change review board so it can understand why there is a problem on a project and make sure they do the analysis properly so the problem does not happen again. This idea translates to future programs: projects need to have stronger requirements flow and stronger oversight through all phases (requirements flow is from aircraft to subsystem to component.)	Hopefully any mistakes made on a project can be rolled out as a lessons learned. It could be a useful tool in future discussions because it would show the consequences in cost and schedule for not doing things a particular way. It would also give systems engineering better standing to get management’s attention.
2	If a design does not meet certain requirements, it could indicate that a part of it is over-specified. If there is margin in the design, then change the requirement and make the paperwork match the design. It is a different story if the design actually needs to be changed.	The part has to go through analysis. Engineers have to analyze the design through various aspects, such as the stress environment. Everyone system needs to be cognizant of their physical, mechanical, electronic, and control interfaces with the other systems. If a change is proposed, they have to consider, “how does the design change affect other aspects of the design?” In a new design, there are many integration meetings. As a specific engineering change or original drawing release is processed by a manager, however, it is unclear whether the different managers or design functions have actually talked to each other. The respondent said that has happened many times and leads to rework changes.
3	Truly understand what the root cause was, and do not jump to conclusions.	

Session	Respondent A	Respondent B
4	Pull the affected parties together, maybe one group missed a requirement or an interface interaction and they have to redesign. Then the groups discuss solutions (i.e. conduct trade studies) to get a best answer. Then get people to analyze it (people most appropriate for that specialty), and get internal and external permission. Formal change process makes the change official by getting change permission from the customer, in contractual or acceptance form. Think of a Corvette; changing the engine, you're still stuck with the transmission, the frame, etc.; the basic design remains the same. When the requirements specify making this change, some engineers believe the design's legacy is not going to change, but then there is no compatibility because one project changed and the other did not. The ones that did not implement the change did not realize the actual full impact of the non-change. Putting a new engine on an aircraft is sometimes a 3-year design effort. There is definitely a slowness to change management. Once the subsystem gets to a certain point, one group owns it and no one analyzes what could go wrong when all the subsystems interact. No one reviews the system as a whole and any little problem becomes problematic.	From a systems perspective, that sort of change is exceedingly rare at the customer level. We don't do a lot of new design. The respondent's systems engineering is based on existing systems that are modified to create a new set of requirements. Given that they exist: they already work. Major design changes are rare.
5	When project managers identify that there is an issue, they come to systems engineers and ask for help to figure it out. They then sit down as a team and do an integrating event, involving all the major players to determine what needs to be done. There are also budgeting issues and the team has to figure out where the money comes from to fix it. It is exciting because the systems engineers get to do with "real" engineering work, in the nitty gritty of the technical aspects to redesign the part or whatever is causing the problem. Now, project managers involve systems engineers from concept to installation. There's a fast churn on the international programs because they want their stuff fast. They can have smaller lot sizes.	Design changes occur with great pain. This company never has enough slack time in any schedule to redo something, so something has to give. 10x philosophy; when you make a change it gets 10 times more expensive with a given time increment. You have to have people work overtime and redo a ton of work, especially later in the schedule. When they encounter a problem they do a root cause on why it happened. The international customers are sometimes pulling old aircraft out of service and need the new ones on-time. There is no wiggle room because they need those aircraft. The international customers are allies to the US and they rely upon not just this company but the US government for support.

Table 5.3: Shortfall the respondents encountered
(researcher-assigned cause codes are in bold below each statement)

<table>
<tr><th>Session</th><th>Respondent A</th><th>Respondent B</th></tr>
<tr><td rowspan="2">1</td><td>A requirement on a project said that if personnel did an inspection, there was a maximum amount of time required to disassemble the system enough to gain access and perform the inspection.

The respondent raised the questions: “How often do you need to inspect that? Is there a periodic inspection?” There was no means to inspect the area or give maintenance access to the area.

Parts were on order for manufacturing and the design was finalized, so this problem required a significant rework.

The engineers needed to redesign the area completely to accommodate an inspection hatch. The project was at risk of cancellation because of this issue.

Contributors: Miscommunication and arrogance</td><td>The company had an agreement with a customer to integrate an engine on to an aircraft. The engineers did not understand the implications of doing that because they did not find out what the requirements affected were. They should have figured that out before they started.

The company had a customer that had a high technical content aircraft. The aircraft was complete and was in testing. The customer disagreed with the method the company had used to prove design success, and introduced requirements for additional tests. There was a big disconnect between the customer’s requirement verification expectations and the company’s. The company had to create a detailed verification matrix over the course of many months before the customer accepted the aircraft. The engineers did not start the job by asking how they were going to verify it.

Contributor: Management overconfidence</td></tr>
<tr><td>Conducted poor requirements engineering

Inadequately communicated</td><td>Inadequately communicated

Conducted poor requirements engineering</td></tr>
<tr><td rowspan="2">2</td><td>The respondent did something and wiped out half of the DOORS links to the program. He worked out the solution, fixed it and it was done. The interaction of what he did versus what the effect on the system as a whole was what caused the problem and learning that insight was valuable moving forward as new challenges or changes were encountered.

Contributors: Not knowing that that task that he did had that unintended consequence. Once it was raised to him, it just took a little time to figure out what happened and implement the fix. Another engineer noticed the problem right away, and it was a best-case scenario because he learned something and the problem didn’t linger.</td><td></td></tr>
<tr><td>Failed to consider system interactions in design</td><td></td></tr>
</table>

<table>
<tr><th>Session</th><th>Respondent A</th><th>Respondent B</th></tr>
<tr><td rowspan="2">3</td><td>There was a test failure that took them a while to figure out; it was something they had not thought of.

Contributors: The engineers did not account for that condition in developing loads and stresses.</td><td>A document went out to a customer without the knowledge that it was supposed to verify a requirement. In DOORS it was tied to the requirement, but there was no traceability that it met the requirement. This was a setback through systems engineering where something fell apart and the author of the document did not know he was trying to verify the requirement. There was no traceability.

Looking through the links in DOORS, the document was completed but the engineers never actually proved it to the customer.

The engineers implemented the compliance form that links the item in DOORS to ensure that requirements are captured.

Contributors: Not all people on the project have DOORS access so they rely on others to give them that information. These people have to ask for a compliance form.</td></tr>
<tr><td>Failed to consider changing environment in design</td><td>Kept poor records</td></tr>
<tr><td rowspan="2">4</td><td>Engineers on a project assumed a new system used the same data format as the previous system and although the new system was able to interact with the other systems, the interaction resulted in incorrect actions. They found that they had to analyze the interaction with the new system differently than the old one.

Contributors: The engineers did not document lessons learned very well but they could have avoided repeating that scenario because there is tribal knowledge on that now.</td><td>There was a design that contained an external load: when the load was released, the band holding it would swing down and damage the aircraft. That required a substantial redesign effort. They found the problem through testing.

Contributors: Lack of imagination. The engineers had not considered what will happen when this thing releases. The respondent suggested using FMECA.

This is a real systems engineering challenge because two completely different subsystems interact</td></tr>
<tr><td>Failed to consider system interactions in design

Did not learn from failure</td><td>Failed to consider system interactions in design</td></tr>
</table>

Session	Respondent A	Respondent B
5	On a project, certain things were put in certain areas to make them easily accessible to the user. Boxes and switches need to be installed that drive where things are put on the console. It’s been a huge integration challenge. Anything that comes in direct contact with the user can be challenging. Risk management is a big problem. The company is pushing more risk management upfront during the proposal phases. In the past, engineers would do the risk management as the project progresses. Now they’re doing more in-depth and robust risk programs. For instance, when the project proposal is due. Risk mitigation planning during the proposal phase is helpful because it also helps with the master schedule. If the engineers identify any risk they can plan that into the schedule and give themselves time to deal with it. Design work and dollar amounts are associated now with risks and that makes the program more successful. However, risk mitigation planning has also been challenging because projects are spending more budget doing that at the onset without proper and clear direction/process changes to accommodate. It takes a week’s worth of risk meetings. They want to win the proposal and eventually make that program.	On a project, the location of a switch was not in a correct place on a subsystem. Human factors requirements were violated to keep it there, even though the customer wanted to keep it there and pushed for it to be located there. The more requirements you do earlier (even during a proposal) becomes hugely important. Get more upfront requirements work done as early as possible.
	Managed risk poorly	**Failed to consider human factor** **Conducted poor requirements engineering**

Table 5.4: Phase where they encounter the most shortfalls

Session	Respondent A	Respondent B
1		Planning in the very beginning is a problem because this company does not plan well.
2	Interfaces are always a problem and the engineers at this company spend a lot of time on interfaces in general.	Interface integration is a big problem because the design of a subsystem needs to integrate well with other subsystems.
3	Most problems show up during requirement verification. That’s when they find shortfalls	Engineers will discover problems during testing because what they designed was not sufficient. However, it is difficult to identify these problems before that point during verification.

Session	Respondent A	Respondent B
4	Problems occur in the proposal stage from not analyzing interactions and getting support among teams in putting forth the effort necessary to get through development.	In the production phase there small changes have a big impact. If an engineer sees a better way to do something at the production level there may be a big impact to cost and schedule. The proposal phase is very important because there are ramifications further on in the design cycle.
5	Production is pretty efficient. No specification is delivered with the proposal. Engineers make a lot of assumptions because they did not do specifications upfront. Issues result from that because after the proposal is accepted there is no time to ask the customer what they want and develop more requirements.	The Navy handles the “right side of the V”. This company just supports them after testing. Most of the shortfalls occur during development and proposal work. “Way at the top of the V”.

Responses Pertaining to Systems Engineering Education

Table 5.5: Academic background and systems engineering training

Session	Respondent A	Respondent B
1	This respondent has a full-time Bachelor's in mechanical engineering, and a Master's in management of technology. He did not undergo additional training to be a systems engineer	This respondent got his Associate's (mechanical engineering), Bachelor's (mechanical engineering) and Master's (management of technology) all going part-time. None of his degrees are in systems engineering and he did not undergo additional training to be a systems engineer
2	This respondent has an electrical engineering BS and MS. He did not undergo additional training to be a systems engineer, but has taken DOORS classes.	This respondent has a BS in Mechanical Engineering and an MBA.
3	This respondent has a BS in Aerospace, MS in Aerospace, and an MS in system design and management. He has no formal systems engineering training. As far as the curriculum for his Bachelor's, there was no specific material relating to systems engineering. In his MS there was one course. Most of his specific systems engineering knowledge comes from being a pilot since he also did maintenance. He did some company-held courses, but mostly he has done a lot of on the job training.	This respondent has a BS in mechanical engineering and an MS in Management of technology. For systems engineering, he has received on-the-job training.
4	This respondent has a BS in Aerospace and an MS in systems engineering. He says his degree was technically called an Industrial engineering degree but this was before the university was allowed to call it "systems engineering." He also has an MBA. He was a Navy pilot (this included ground school and flying), so he has experienced use of the end product and knowledge of the systems.	This respondent has a PhD in a biology field. He got involved in modeling systems, and learned computer programming so he wound up with the software group. For systems engineering, he works just based on his experience, but he has attended occasional workshop-type courses and training on how to use systems engineering tools
5	This respondent has a BS in Aerospace Engineering. For systems engineering, she's done mostly on-the-job training, with a majority of the training she received coming from the group responsible for system engineering processes.	This respondent has a mechanical engineering BS degree with a Master's in management and a focus in product development. As far as systems engineering, he has learned a lot on-the-job, but there was training run by their systems engineering process group, which is responsible for organizational processes and training

Table 5.6: Areas systems engineers struggle with

Session	Respondent A	Respondent B
1	There is a fine line between defining aspects at the aircraft level and the subsystem level (this is also a problem programmatically). It is hard to define the line, and also hard to define what the systems engineering scope is. Systems engineers' roles bleed in several directions. New systems engineers get dominated by their program because they do not understand what the scope of their job is. New systems engineers lack training and guidance.	This respondent has two new employees right now. They struggle with clarity of communication; trying to communicate an idea unambiguously (like a risk or requirement). They need to define what they talk about in a way that cannot be misinterpreted and can be acted upon. He does not think they've been trained in clear communication because school teaches students to write things deliberately so their statements encompass a lot of ideas. That way the students can be correct no matter what when talking to a professor. It is a different environment in industry because clarity of communication is required to define the precision of the work. Another problem is that people usually give ambiguous work (i.e. work that is not readily assignable to a clearer area) to systems engineers because what the systems engineers do in the company is not clear to others.
2	New hires have trouble with understanding the big picture. There is a stovepipe mentality that comes from new hires (fresh out of school): "I'm a mechanical engineer, I know this." Process and requirements work is tedious and anyone will struggle with it without the right mindset.	New hires lack knowledge and experience across different disciplines of design. They are not familiar with what the product is. For instance, a new hire may have a mechanical engineering background, but being a systems engineer requires them to know about what electrical engineers need for their requirements. They may not know how to do the design work but they need to know what the designers consider, like environment, interfaces, signals, and structural loads (whatever the hardware or software is exposed to). Systems engineers have to know to ask the right questions. This skill takes experience and being around for a period of time.
3	New hires do not understand this company's product. An understanding of aircraft, including their intricacies, coordination, and interfaces is needed. New hires' academic background is lacking in this regard, especially because this company has unique implementations and each program does everything a little differently.	New hires are not always familiar with the processes that interconnect and tie together (i.e. risk management, requirements management). New hires need to understand those and how they link together. There is a uniqueness to the work at this company due to customer demands.

Session	Respondent A	Respondent B
4	This respondent thinks new systems engineers have a lack of experience in the system as a whole. A single project at this company has so many different subsystems and subsequently how they tie together overwhelms new hires. Since this respondent was a Navy pilot, he already had first-hand experience with the product and this helped him have a much smoother transition than others. The systems produced by this company are complex and the new hires do not have the product experience. Understanding the system is difficult and even if they have the methodology they do not understand the interrelationship and functionality of the parts	New hires do not understand what they're looking at when they see a requirement. The company has a lot of inexperienced people in general writing requirements. They can over-write (include too much detail) or under-write (include not enough detail). On the other hand, there are a lot of poorly designed requirements from the customer that newly hired systems engineers do not recognize are poorly designed. The requirements from the customer can be a statement of work and trying to translate that to performance requirement means it is untestable. In general, new hires don't get exposed to requirement writing. The writing discipline is not emphasized in engineering programs, nor is the interpretation of the requirements.
5	New hires struggle with people skills and the ability to "own everything and not own everything." Systems engineers have a lot of disciplines they are responsible for but no one reports to them. They have responsibility but no authority. Being able to work as part of a large team is important for people who are in school to understand. Each person in a team has different skills, different functions, and different responsibilities. When this respondent was first hired, her platform was unfamiliar with systems engineers and saw them as in the way. People need to see the value of systems engineering.	People are well-trained and versed in systems engineering through school. However, the ancillary skills come with experience. New hires need help with road blocks when they encounter something that prevents them from doing what they need to do. Not everyone understands what systems engineering is and what they do (especially outside the Avionics world)

Table 5.7: Traits in new hires in systems engineering

Session	Respondent A	Respondent B
1	A new hire needs to know about system attributes and how every system works. They need to shoehorn in what the customer wants and make it work within the organization. This respondent looks for whether a candidate can demonstrate the ability to focus and use the top-down approach. He wants to know whether the candidate has the ability to see the path ahead, and to execute the program in an efficient way.	New hires should demonstrate flexibility and adaptability. The role of systems engineer is so varied and so broad; new hires have to be able to accommodate a variety of expectations and demands. Systems engineer is a job for a generalist, not a specialist.
2		

Session	Respondent A	Respondent B
3	This respondent looks for new hires to have a basic understanding of this company's processes and have a good academic foundation. He also looks for people being able to work in and interact with a team.	New hires should have an open mind because some people focus on a single components. They should look at the system from an overall perspective and know how aspects interact with each other at a high level. Even if new hires don't have the ability to do that yet, they should have an openness to learn that.
4	This respondent looks for people who are aggressive, assertive, and confident. Systems engineers have to work with other people, organize them, recognize potential problems and get people to work together. These people bring separate teams together to resolve situations and know how to do that. New hires have not dealt with these aspects of a job before. It was a big change going from working solely in a single discipline to a systems engineering role that supports many different programs. System engineering leads on my projects have to act in a more managerial role to the subject matter experts and also be supportive to the chief engineers and program managers. Many times, we are the glue to bridge those two groups.	New hires should have creativity and independence.
5	This respondent wants new hires who drive tasks to closure, especially because systems engineers have to ensure deliverables to customers within specific time frames. If the team does not make the deliverables accurate, correct, and on-time, there may be a penalty. It is important for the team to take ownership, especially for setbacks and to not blame others for problems that occur. Adaptability is another good trait. Systems engineering involves an ever-changing workload and pressure from management. New hires need to find time to complete tasks within tight deadlines Dealing with various personalities of people on the project is another aspect of systems engineering. There is a lot of background "stuff" that systems engineers do to make things look flawless.	This respondent looks for a strong technical background, preferably with experience in requirements. He also looks for a sense of ownership and driving tasks to closure with accuracy and timeliness. This company currently uses behavior-based interviews, where candidates are asked to describe a situation where they have had to solve a problem or perform some other task and describe what their specific input was, and avoiding generalities. Working well with a team is another trait he looks for because systems engineers have to deal with different personalities and take on a leadership role to pull together thing like design reviews. A lot of information is not in a plan anywhere, so people have to stay on top of things.

Table 5.8: Traits in respected systems engineers

Session	Respondent A	Respondent B
1	[Same response as previous question]	Highly respected systems engineers have integrity. These have to be the people who identify when a system does not work. They have to deal with rocks being thrown at them and still stick by their guns.
2	Systems engineers need to be able to relate various pieces of a project together. Successful systems engineers have the ability to multi-task. These people are able to have 35 things going on and not drop anything. There are only a few individuals that are able to do that without being stressed. That is not a trait or a skill that is easily transferred to new people. That cannot be trained, although it can be improved upon. A lot of it is personality driven.	His previous company was far more advanced in systems engineering than at this company. This respondent learned a lot from the more experienced systems engineers in his group. They know how everything fits together. People who learned different aspects of the business before becoming a systems engineers have useful experience.
3		Respected systems engineers have a willingness to train others. They have the knowledge and share it with others.
4	Leadership is a characteristic that is respected across engineering. However, systems engineers have to be more team-oriented earlier in their career. For the most part, this company gets people into systems engineering from other skill sets when they exhibit those capabilities. For people coming from school it is difficult in an interview to tell whether they have those qualities.	[Same response as previous question]
5	This respondent respects calmness in other systems engineers when things are “blowing up left and right”. These people have the attitude of “it will get done, we’ll get it done.” She doesn’t want to see the leader unravelling in front of the team. Good systems engineers mobilize their workers without being panicked. Great systems engineers practice calmness and have good leadership qualities and have strong technical ability. Some of the people she has worked with in a leadership capacity understand the material very well and are able to guide the teams better because they have the background experience.	

Table 5.9: Changes to systems engineering practices over time

Session	Respondent A	Respondent B
1	Systems engineering as a practice in this company is quite young. In the past systems engineers had a hard time of selling the idea of tracking and maintaining requirements. Systems engineers at this company get to watch the evolution of systems engineering practices locally and also globally. There is a global approach to organized processes and an application of metrics that was not present before. Previously systems engineering efforts had been done in an ad hoc way, that made the process take longer than it needed to. The Navy wanted to do systems engineering processes better, and that was the start of the shift.	The change in systems engineering practices is that the company now has them. These practices did not exist when this respondent started working at this company 30 years ago. In the beginning systems engineering did not exist. Expectations were simpler in the beginning. Often, customers did not explicitly define what they wanted. There was a lot of back and forth and it was a messy process that cost both parties money.
2	There is now more consistency in systems engineering for big-picture ideas. Most of the DOORS admins are consistent but some people are not admin but are doing an admin-type job. For example, a new program struggled with the DOORS infrastructure because they did not appoint an experienced DOORS admin. There is a trend at this company to use more DOORS, so more people know how to use it. At first this respondent was responsible for training a few people, but now he trains people at the whole company in how to use DOORS. In general, the technologies available are changing and people are becoming more familiar with it.	People can spend days reading process procedures, and they end up not using them. People know where to go when they need details, but they don't read the full report. When people go from one program to the next, for a given discipline there is consistency, but there is none for systems engineering. There are different things that need to get done based on what project a person is on. Processes are different given any different project.
3	The company uses systems engineering practices more than it did in the past. There is also a broader implementation of systems engineering practices.	Upper management awareness has changed in that now they realize that systems engineering is necessary for projects to show compliance to their customer. This has led to more acceptance of how the systems engineers handle requirements and an improved "paper trail" for changing requirements. Ownership is now placed on upper management, rather than systems engineering when dealing with compliance. This makes them focus more on improving their own requirements because they own them.

Session	Respondent A	Respondent B
4	At one point in the past this respondent had tried to convince a manager that he needed a requirements management tool (i.e. DOORS, before DOORS existed) for use on a particular program, and the manager was adamant that he didn't want those fancy tools ("phony attempts to replace real engineering"). Using these tools has become much more accepted and expected because the customer has demanded it, especially the more technologically-savvy customers.	
5		There has been a ramp-up of the importance of systems engineering in general. As the company changed, the focus changed from engineering-focused to manufacturing-based. Now it's "put them together, get them sold, get the revenue" It used to be all about all the engineering. However, a manufacturing ramp up led to a systems engineering input ramp up. All the people involved in a project used to be collocated. These people did everything in the same place and had the same focus. As that went away, it brought the need for systems engineering to pull it all together and make it less scattered. Systems engineering is the glue.

Responses Pertaining to Systems Engineering Tools

Table 5.10: What tools the systems engineers use

Session	Respondent A	Respondent B
1	He draws the Systems "V" on his white board once a week. It's a global way of understanding where his team is in the process, but it doesn't really drive what they do. Tools he uses include: Block diagram, peer-review processes, requirements tracking and flow tools, and risk management tools.	He works at a more granular level than would be useful for the tools Respondent 1A used. He doesn't work at the big-picture level that he would have to use the Systems "V" or other general tools. His team uses risk management tools and tools that "connect the dots". He says he doesn't spend a lot of time using these tools and that his team does not use a tool formally.
2	This respondent uses DOORS, process change requests, and Trade studies	This respondent uses engineering instructions when he develops test articles. He also uses acceptance test procedures, and he specifically uses DOORS for requirements flow-down.
3	This respondent uses requirements analysis tools.	Some of the specific forms and processes this respondents uses are company-specific. These processes allow engineers to review requirements and check how each meets the corresponding specification. More generally, he uses risk cube for risk management, trade Study methodologies (which is consistent across all platforms), and DOORS for requirement management.
4	This respondent uses functional flow block diagrams, operational sequence diagram, use-case tools, and timing analyses (complex or simple). He also uses requirements management tools for traceability and flow down of requirements and tests. More generally, he uses Matlab (and other mathematical tools) and spreadsheet tools. Some spreadsheets are huge, and he uses databases to keep track of things with large amounts of data. This is difficult to do because you can't necessarily trust the spreadsheet tool, especially when tracking document changes from multiple users.	The main tool this respondent uses is DOORS, as a way to manage requirements. DOORS is very old and it is not well-integrated with other system engineering tools. A more integrated system would be a huge benefit. For instance, he does a lot of "timelines" and uses Microsoft project that lets him figure out the critical path between things. This software is not integrated with DOORS. There are way too many manual engineering processes.

Session	Respondent A	Respondent B
5	This respondent uses the Systems "V" a lot because it's a standard tool in the industry. The "heaviest" tool she uses is DOORS, for requirements. She also uses Risk Tool to manage risks; it's an internally generated database tool. In that tool her team manages risk for the entire program and other things like the document structure. All of the documents her team uses have to be processed through the Risk Tool. They use the Systems V as their guide to get to that point. Microsoft, PowerPoint, Excel are other tools they use. They make a ton of presentations for each of their reviews. Coordinating that is challenging.	This respondent uses DOORS and Visio.

Table 5.11: Tools that would have been easier to learn in school

Session	Respondent A	Respondent B
1	This respondent is very far along in his career so learning about a tool in school would not have been useful for him (they would be out of date by now).	This respondent had some exposure to Monte Carlo in school. Once out of school, he had to figure out how to use company-specific tools and implementations.
2	None. This respondent believes that actually using a tool in a career is more useful than learning about it in school. For example, a DOORS class would not be as useful as sitting down and using this program. In school, students learn a "base language" that helps them understand the architecture of programs. However, these types of courses don't go into enough detail to help students create a DOORS architecture. Taking a class that you're not going to use in your day-to-day activity is not conducive to your time.	None. He has been out of school too long to find it useful. Digital 3D modeling of parts (Catia-5) would have been useful to learn in school.
3	Some tools like timing analysis and other analytical tools would have been useful to learn in school.	DOORS may have been useful. People use it for the first time when they are hired to the company. He didn't have any exposure at all, even in his masters. There is training in-house, but it would be good to come into the job knowing about that
4	Conceptual tools are relatively straightforward so the academic setting is a good place to pick up on them. Requirements traceability would have been a good one to learn in school. The implementation of the tools have some applicability for an academic setting (at the advanced level, not at excel level).	No, in fact it's the opposite. Most of the tools that he learned in an academic setting he already had experience with. Generally systems engineers know what problems they need to solve and what they need to use to solve it. He doesn't think anyone should waste time using a generalized tool with no specific problem to solve.

Session	Respondent A	Respondent B
5	This respondent said DOORS because it's the industry tool for requirements management, she would have liked to at least get exposed to it in school. Once she started using it, it was like Excel "on steroids." Everything else systems engineers use are tools that help with Earned Value (EV: measure performance when working on a program). She's also responsible for budgets and activities for all the people that report to her. She works with 4-5 projects and each has its own budget, schedule, and systems engineering duties and she has to report at the end of the month for every single open activity. She has to evaluate whether they need more people, budgeting, etc. at the end of every month. She would like to see more management coursework in school. She's found that the new engineers focus on the technical aspect but she does less technical work now (more management than technical work). New people coming in to this company get exposed to more project management tasks now. Project management training would be extremely useful (concept to delivery type learning). Some of the programs she works with have new hires doing more project management (even fresh out of school) She says a lot of people on her level are groomed to be chief engineers and program management Systems engineers always gets taken into one of the higher-level jobs (groomed for management).	Any systems engineering curriculum in college should start with an idea and turn it into a part. Students should see the design from the requirement through to the end of the product lifecycle. Students these days are well-versed in Microsoft products (Word, Excel, and PowerPoint). At his level, systems engineers inherit more and more activities and duties. He thinks people getting their MBA would get a lot of those management skills he uses.

Table 5.12: Tools the systems engineers find useful

Session	Respondent A	Respondent B
1	This respondent likes the contract operating system tool because it lets him dig into management better. This tool has an incredible amount of program history: estimating and cost tracking are easy to do. There is also an internally-generated database tool. This tool tracks technical issues and risks. This tool has a great interface, and it is simple and straightforward. Most of the program information is in one place. The internally generated database has an inconsistent use between commercial and military programs. Someone coming from the military side can easily learn how the commercial side uses the tool, and vice versa. But there is not one way to use the tool. Users have to spend time learning if they go from military to commercial or vice-versa. Commercial has a high number of customers that have few aircraft. They are all tracked under one contract. They do things to improve the product, paid for by this company. Military has few customers with a high number of aircraft. They don't do anything the customer doesn't pay for. He has network latency issues and interface issues through his personal computer. Sometimes information is hard to find. There is tribal knowledge about how to make the tool work.	This respondent likes the risk quantitative analysis tool and trade study tool. They are not hard or complicated but he has to spend a little time with it. Risk quantitative analysis: provides insight into the programmatic impact of any number of risks. Risk used to be a single entity that has an impact like cost or schedule. Now we have all these risks that have cost impacts and different likelihoods. What it means the risk to the overall program is much less visible; this tool allows you to visualize the potential impact of the summary of the risks. The quantitative analysis tool is an Excel spreadsheet running on the user's local laptop that sometimes slows the laptop greatly. It should be hosted on a more capable device, like a company server. DAR: lets the user take the subjectivity out of a trade study, which is helpful because engineers come in with a presupposed result to the trade study. If the user uses the DAR correctly, they can eliminate their prejudice from the results and justify it quantitatively and substantiate it. This tool has no real flaws other than the user needs to get "used to it". It could have a better interface.
2	DOORS is a useful tool because it has co-location of material. A user can export out to other formats, and this company has done an excellent job of generating specific scripts or tools that work with DOORS that allow one to create things that it wouldn't be able to do on its own (e.g. exporting out to Word document). This company has a lot of these tools that enhance the use of DOORS as a whole. Even if you don't have 100% of the team working in the tool itself, you can export information out to Word and allow for crosstalk. Additionally, their customer has full read access to DOORS so they can see everything. It would be better if everyone was trained to use DOORS, but they can get from A to B with little difficulty. Customers specify in their contracts that they want full read access but don't necessarily use it. You have to get people to actually use it.	This respondent agrees that DOORS is a useful tool. He wishes everything he did relating to requirements and verification and documentation was done in DOORS. Not everyone is familiar with DOORS. It would be a single place to input information. Half the people know how to use it. So you export out of DOORS and adds more manual steps. If it was used entirely you would have documents, test plans, and test reports in DOORS.

Session	Respondent A	Respondent B
3	N/A	DOORS is a useful tool. However, the tool is not the most user-friendly and a little overwhelming, but users get used to the flaws over time. There are intricacies in linking models to keep traceability of requirements. Traceability becomes cumbersome. He's not sure if this company has tried to find something better (i.e. more user friendly).
4	Something that is dramatically underused at this company is the context diagram showing what the interfaces are, putting clear distinct definitions on those interfaces, confirming and using those interface names, and referring to that at a later point in the project. This tool provides a clear naming convention for the components and interfaces so you don't have different groups referring to a subsystem as sometimes having different groups of components (different subsets) and what the interfaces are. Someone may think loosely about a definition. Different people think in different ways. There are limited mechanisms as defining those tools in implementation. He would like to see an implementation tool that manages and defines those naming conventions for subsystems; keeps naming conventions consistent. If you write the specs in Word or DOORS it doesn't tie together the conventions.	Really useful: "top-down" was on a floppy disk. You could do a flow diagram, if you double-clicked on a box you could make a whole flow diagram. No other tools have had this feature since.
5	N/A	DOORS and Visio are useful tools. Visio is a basic tool that is good for flow charts and spec trees (they work with very complicated spec trees that involve 3 different industry partners). It can be very useful in creating graphics demonstrating that you have a good grasp on requirements flow-down. He thinks that if users criticize flaws in the tool, it shows a flaw with the user. Users have to figure out how to use it.

APPENDIX C. STUDENT SURVEY

Levee Question

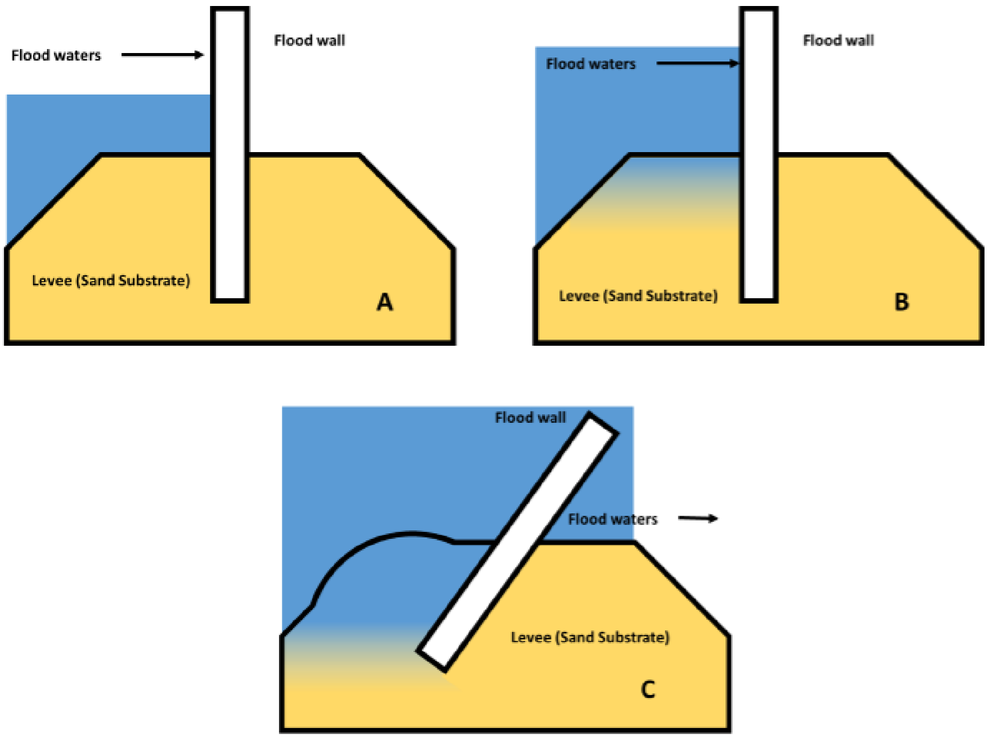


Figure 5.1: Flood Wall question diagram

Figure C.1 shows a simplified diagram of a flood wall that prevents water from overtaking a levee. The flood wall is built in a sand substrate, which may become saturated with flood waters and lead to the flood wall tipping.

Of the design principles in Table C.2, which do you think this design satisfies?

If you wanted to improve this particular design, which design principle do you think is the most important to satisfy and why?

Table 5.13: Grading scheme for Levee question

#	Quality	Explanation/Example
General response qualities		
1	Student's response contains more ideas than those given in the question description. Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].	Responses ONLY containing these ideas do NOT have this quality: • The water saturates the sand substrate • The wall tips over when the sand becomes saturated • Flood water overtakes the wall The student's response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality.
		Example of a response with an "extra" factor: If you had room, have a marsh-like nature as a barrier in between the levy and people/property. Other examples include ideas like what other risks occur when there is a flood such as landslides and debris carried by the water.
2	Student's response reflects cognitive competencies that successful systems engineers possess. Justification: Systems engineers have certain cognitive competencies [Frank, 2012]. Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].	The student's response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question. Typically the cookie-cutter, bland student responses do not display this quality. The student's response must demonstrate that they thought about the problem in a "systems engineering" way.
		Understand the whole system and see the big picture: "understand the whole system beyond its elements, sub-systems, assemblies and components, and recognize how each element functions as part of the entire system." Example: A levee is usually a very large system, and if any part of the levee partially fails (regardless of how structurally sound the rest of the levee is), the system as a whole fails. Therefore, layered redundancy, such as a water pump, could prevent propagation of localized spill water or, at the worst case, prolonged water damage.
		Understand interconnections: "understand the interconnections and mutual influences and interrelations among system elements." Example: Improve the performance of the system by implementing a means by which the substrate and the wall act in concert, like the wall trapping water so that it can effectively drain through the substrate rather than saturating it.

#	Quality	Explanation/Example
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems." Example: Incorporate a hazard barrier such as a membrane between the sand and the water that will prevent the levee from saturating without substantially changing the system.
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level." Example: Find a safe way to displace the water in addition to the wall which prevents water from flowing over the levee. Diversifying the method by which the water is safely contained would increase the number of modes by which the system would have to fail for the water to cause damage.
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system." Example: a dike to divert the water away from homes. A layered redundancy would be susceptible to the same flaws as the primary layer of protection and could therefore also be overwhelmed quickly, and some of the other options are not feasible such as localized functionality (The wall must be at that location) and physical redundancy (it would be difficult to put in a replacement wall while the area was flooded, even if one was ready).
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	**Able to take into consideration non-engineering factors** Example: It's a levee system that can tip over and completely fail if there's a flood. Not only does it fail at its purpose, but it provides local residents with a false sense of security. In some ways, using this flawed design is worse than having no levee at all, because residents will assume the levee was well-designed when in reality it was not.
Question-specific response qualities		
4	Student identified a design principle that the system satisfies ("absorbs damage", "localized functionality", or "hazard barriers").	The student ticked a box for design principle(s) that the system satisfies, indicating that the student believes that the design as-is satisfies this principle in some way. The three that apply to this system are "absorbs damage", "localized functionality", and "hazard barriers". Ticking any one of these three satisfies this quality.
5	Student identified a design principle to improve the system.	In their discussion, the student identifies a design principle that the system does not currently satisfy and could be used to improve the system in some way. This quality is NOT satisfied if the student "ticked" the box for a design principle and then selected the same design principle in their discussion (this indicates a contradiction).

#	Quality	Explanation/Example
6	Student's discussion matches their design principle. "Consistent"	The student's response indicates that they **understood what the design principle meant** in the context of the question. Example of a discussion that does not show how the principle could be applied to the system: *I would pick to improve upon the principle of physical redundancy because the system can be made so that the flood wall is thicker so if the wall does tip, there is still a significant area preventing flood waters from passing.* How would making the flood wall thicker improve physical redundancy? That does not make sense in the context of this design principle and does not demonstrate that the student understood what the design principle meant. Note that this response does satisfy qualities 7 and 8.
7	Student's discussion indicates how the design principle is applicable to this system. "Indicates apprehension of design principle and how it applies to the system"	The student's response indicates that the student fully apprehended the design principle and **how the design principle would apply to the system**. The discussion specifically mentions some idea that would apply to the system. Example of a discussion that does NOT show how the principle could be applied to the system: *Physical redundancy. This design does not have any redundancy for me.* The student picked a design principle but did not provide a clear example showing how the design principle applied to the system. Note that this response does not satisfy qualities 6 or 8 either.
8	Student discussed how the design principle could improve the system. "Would improve system"	The student's response describes **how the design principle would improve the system** in the context of this question by preventing the failure, make the failure not as severe (e.g. limiting damage or risk of injury/death), or make the failure not as frequent. Example of a discussion that does NOT show how the design principle would improve the system: *saturation of sand with water is a hazard and including a hazard barrier in the design principles will increase reliability.* The student claims that this design principle will increase reliability but does not describe how this would increase reliability. Note that this response satisfies quality 6 but not 7.

Oilrig Question

Table 5.14: Design principles for Oilrig and Levee questions

Design principle	**Description**
Absorbs damage	The system shall be capable of absorbing the magnitude of the disruption that it encounters (e.g. a phone case absorbs shock damage if you drop your phone).
Contains physical redundancy	One or more independent components of a system may fail and the system will still function (e.g. a car has a spare tire in case of a flat tire).
Contains functional redundancy	There should be two or more different ways to perform a critical task (e.g. to prevent sunburn you could apply sunscreen or wear more clothing).
Contains layered redundancy	More layers leads to more resiliency (e.g. if your car breaks down, you can take the bus, rent a car, or ask a friend for a ride).
Contains non-localized functionality	The functionality of a system is not contained to a single node (e.g. if an airport shuts down, other airports nearby accept rerouted traffic).
Contains beneficial interaction	Two or more subsystems interact in a way that actively prevent damage (e.g. elevator safety systems work together to prevent them falling down the elevator shaft).
Contains hazard barrier(s)	A system is protected from a hazard by a barrier (e.g. wearing safety glasses prevents debris from getting in your eyes).

Figure 5.2: Oilrig photo [Thomas 2018]

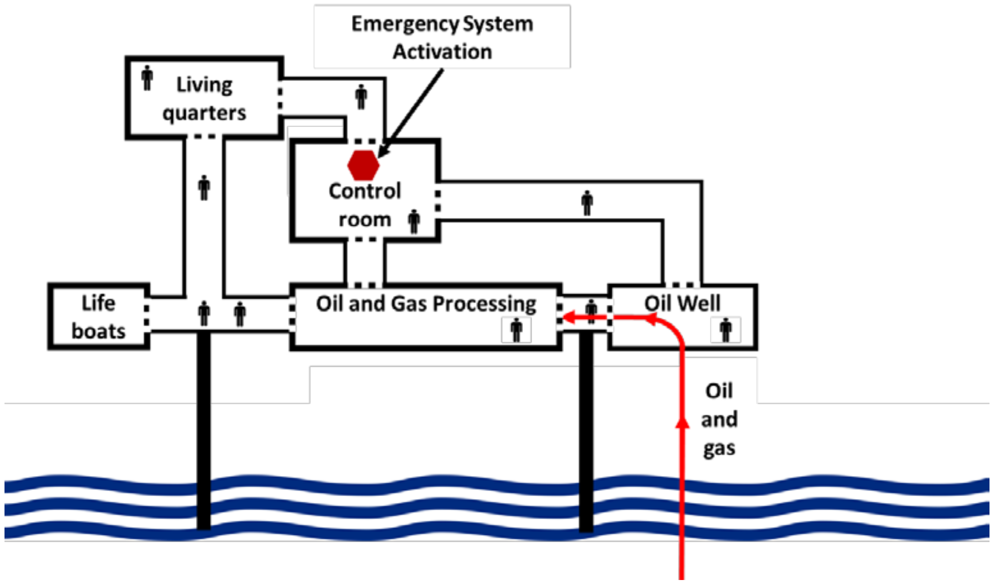


Figure 5.3: Simplified oilrig

Figure C.2 shows a photo of an oil rig. Figure C.3 shows a simplified version of this oil rig. Consider what may occur if an intense fire breaks out in the oil and gas processing section of the rig and starts spreading uncontrollably to the other sections of the rig, including pathways between sections. The instructions given to the crew state that they should await further instruction in the living quarters.

Of the design principles in Table C.2, which do you think this design satisfies?

If you wanted to improve this particular design, which design principle do you think is the most important to satisfy and why?

Table 5.15: Grading scheme for Oilrig question

#	Quality	Explanation/Example
General response qualities		
1	Student's response contains more ideas than those given in the question description. Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].	Responses ONLY containing these ideas do NOT have this quality: • Pathways between sections are a concern for uncontrollable fire spread • There is a single location for emergency system activation • Life boats and emergency system activation are not in the same location as the crew's emergency evacuation location The student's response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality.
		Example of a response with an "extra" factor: I would add redundancy by connecting the living quarters to the lifeboats with a dedicated corridor to give a clear escape path in the event of catastrophic failure. It would result in greater suitability without redesigning the platform entirely. Other examples include ideas like giving people more time to evacuate and how to incorporate design changes to the rig.
2	Student's response reflects cognitive competencies that successful systems engineers possess. Justification: Systems engineers have certain cognitive competencies [Frank, 2012]. Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].	The student's response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question. Typically the cookie-cutter, bland student responses do not display this quality. The student's response must demonstrate that they thought about the problem in a "systems engineering" way.
		Understand the whole system and see the big picture: "understand the whole system beyond its elements, sub-systems, assemblies and components, and recognize how each element functions as part of the entire system." Example: The lifeboats are not attached to the living quarters and getting to the living quarters could require a very long time from certain areas of the rig. If escape routes, life boats, and safe rooms were available near all locations then less time would be required after a fire breaks out.
		Understand interconnections: "understand the interconnections and mutual influences and interrelations among system elements." Example: Since the oil/gas processing area is right near the life boats, this prevents the people in the living quarters from accessing the life boats as they would have to walk right by the fire. There should either be other life boats or at least an alternative route

#	Quality	Explanation/Example
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems." Example: I will add non-localized functionality in this case by adding more points where the crew can get to life boats. As it is really hard to race against damages that may happen on an oil rig, including fire.
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level." Example: if the crew are told to wait in a spot that could block them from accessing lifeboats in the event of a horrific fire, then there should at least be a really good damage control system in place to stop this from happening.
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system." Example: I would add redundancy by connecting the living quarters to the lifeboats with a dedicated corridor to give a clear escape path in the event of catastrophic failure. It would result in greater suitability without redesigning the platform entirely.
		Able to take into consideration non-engineering factors: the student considers non-engineering factors related to the question. Example: If a fire breaks out in the oil and gas processing room, someone has to go from that room to the control room and activate the emergency system. In the time that it takes for someone to travel that distance, oil and gas are continuing to enter the oil and gas processing room, presumably adding fuel to the fire.
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	The grader understands what the student is communicating with minimal confusion or conjecture. This does not necessarily mean that the response is completely spelling- or grammar-error free. Did you understand right away what the student was trying to say?
Question-specific response qualities		
4	Student identified a design principle that the system satisfies ("absorbs damage", "localized functionality", or "hazard barriers").	The student ticked a box for design principle(s) that the system satisfies, indicating that the student believes that the design as-is satisfies this principle in some way. The three that apply to this system are "absorbs damage", "localized functionality", and "hazard barriers". Ticking any one of these three satisfies this quality.

#	Quality	Explanation/Example
5	Student identified a design principle to improve the system.	In their discussion, the student identifies a design principle that the system does not currently satisfy and could be used to improve the system in some way. This quality is NOT satisfied if the student "ticked" the box for a design principle and then selected the same design principle in their discussion (this indicates a contradiction).
6	Student's discussion matches their design principle. "Consistent"	The student's response indicates that they **understood what the design principle meant** in the context of the question. Example of a discussion that does not show how the principle could be applied to the system: *physical redundancy. There is no physical protection if this area has a fire.* How is there not physical redundancy in this system? This response does not demonstrate that the student understood what the design principle meant.
7	Student's discussion indicates how the design principle is applicable to this system. "Indicates apprehension of design principle and how it applies to the system"	The student's response indicates that the student fully apprehended the design principle and **how the design principle would apply to the system**. The discussion specifically mentions some idea that would apply to the system. Example of a discussion that does NOT show how the principle could be applied to the system: *To improve the system with beneficial interaction, having an emergency path from living quarters to the life boats may improve the number of lives saved* The student picked a design principle but did not provide a clear example showing how the design principle applied to the system.
8	Student discussed how the design principle could improve the system. "Would improve system"	The student's response describes **how the design principle would improve the system** in the context of this question by preventing the failure, make the failure not as severe (e.g. limiting damage or risk of injury/death), or make the failure not as frequent. Example of a discussion that does NOT show how the design principle would improve the system: *I would apply the design that contains beneficial interaction because I think this option is helpful* The student claims that this design principle will be helpful but does not describe how.

Boat Race Question

Figure 5.4: The hydroplane, Ellstrom Elam Plus, at the 2006 Madison Regatta [Schneid 2006]

A boat racing team is attempting to decide whether to participate in a lucrative race. However, the team has been experiencing engine failures ranging from minor to debilitating all season and another loss on television would be devastating, but not racing would lose their sponsorship for the rest of the season. So far the mechanics have been unable to pin down exactly what is causing the failures. Since they store their boat outside in the water, one mechanic suggested that cold temperatures may be a factor in engine failure, but the other mechanics are skeptical. The next race would take place on a morning where the ambient temperature is 40°F. The mechanic provided the following graph of engine failures:

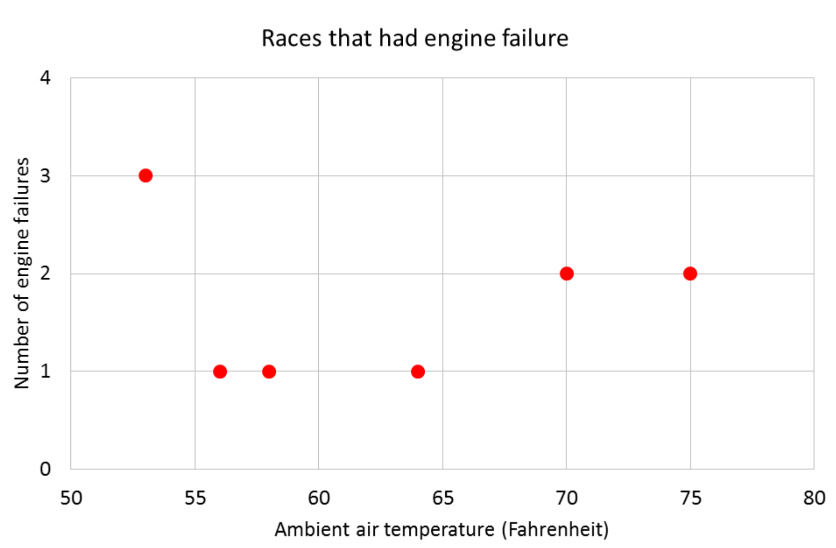


Figure 5.5: Graph of engine failure versus temperature

What other data or factors should the crew consider when making this decision?

Table 5.16: Grading scheme for Boat Race question

<table>
<tr><th>#</th><th>Quality</th><th>Explanation</th></tr>
<tr><td colspan="3">General response qualities</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Student’s response contains more ideas than those given in the question description.
Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].</td><td>Responses ONLY containing these ideas do NOT have this quality:
Aspects related to temperature (weather, water/air temp)
Problems with engine components, like fuel, oil, parts, engine systems, etc.
Problems specifically related to sponsorship loss
If the student only discusses these ideas, it is unclear whether the student is truly considering “other factors” the crew should consider.
The student’s response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality.</td></tr>
<tr><td>Example of a response with an “extra” factor: the reason for boat racing: for fun, or as a job?

Other examples include ideas like how the components of the boat react with water, whether the engine failure is dangerous to the pilot, the quality of maintenance work, and how the pilot was treating the boat during the race.</td></tr>
<tr><td rowspan="3">2</td><td rowspan="3">Student’s response reflects cognitive competencies that successful systems engineers possess.
Justification: Systems engineers have certain cognitive competencies [Frank, 2012].
Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].</td><td>The student’s response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question.

Typically the cookie-cutter, bland student responses do not display this quality. The student’s response must demonstrate that they thought about the problem in a “systems engineering” way.</td></tr>
<tr><td>Understand the whole system and see the big picture: “understand the whole system beyond its elements, sub-systems, assemblies and components, and recognize how each element functions as part of the entire system.”

Example: The team needs to consider the human life involved because if they do not race the team might fail, but all alive. With that said, the person would say go. However, if something happens to the person on TV, lose sponsorships also.</td></tr>
<tr><td>Understand interconnections: “understand the interconnections and mutual influences and interrelations among system elements.”

Example: Severity of engine failures. Which ones were most debilitating or most minor and in what temperature did they occur?</td></tr>
</table>

#	Quality	Explanation
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems." Example: The behavior of the fuel used in the engine under low temperatures. The wear of the engine (i.e for how many hours it has been used).
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level." Example: Relative humidity, altitude/air density. It could be that there is absolutely no relationship with anything relating to the failures, but there are most likely more factors than just temperature
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system." Example: Engine failure due to too many practices, causing overheating on its engine. So, the data on how many times they have heated/started their engine in daily and weekly and their relationships to engine failures.
		Able to take into consideration non-engineering factors: the student considers non-engineering factors related to the question. Example: the reason for boat racing: for fun, or as a job?
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	The grader understands what the student is communicating with minimal confusion or conjecture. This does not necessarily mean that the response is completely spelling- or grammar-error free. Did you understand right away what the student was trying to say?
Question-specific response qualities		
4	Student identified a relevant factor the crew should consider.	The student identified a factor that could contribute to the boat failure or that would help the crew consider whether or not to go through with the race. An "irrelevant" factor is a factor the crew should not necessarily consider when determining the source of engine failure or whether to follow through with the race, or a factor that is not specific enough to make a relevancy judgement on.
5	Student identified more than one relevant factor the crew should consider.	Listed more than one relevant factor, even if they are related to each other.
6	Student-identified relevant factors relate to more than one aspect of the system.	Factors the student listed are from independent aspects of the system. These factors do not all relate to engine components or atmospheric qualities like air temperature/water temperature.

#	Quality	Explanation
7	Student describes how factors relate (or do not relate) to poor performance in the system.	The student demonstrates that they understand how this factor would be related to engine failure or would help the crew make the decision of whether or not to race.
8	Student proposes ideas for how to investigate or fix a factor they identify.	The student proposes either (1) how the crew should verify whether this factor is related to engine failure, or (2) how the factor could be addressed before the boat race.

Aircraft Empennage Question

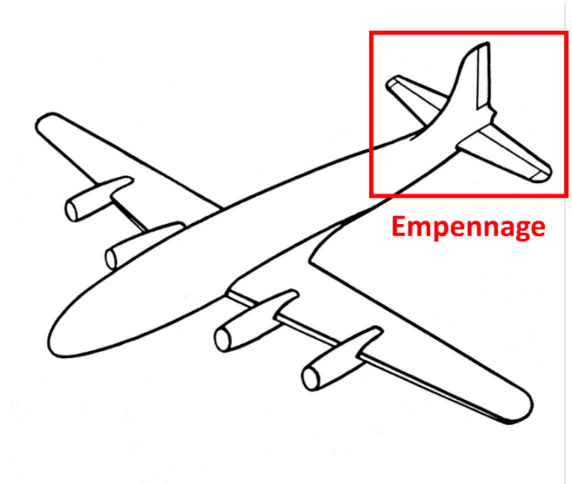


Figure 5.6: Empennage question first diagram

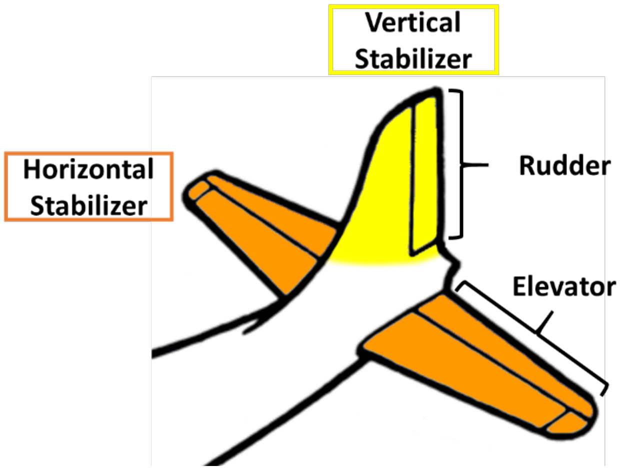
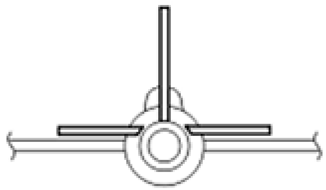
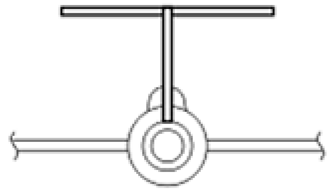


Figure 5.7: Empennage question second diagram

Figure C.6 shows the location of the empennage on the aircraft, and Figure C.7 shows the empennage terminology in a fuselage-mounted configuration. The elevator controls the pitch of the aircraft (up-and-down motion), while the rudder controls the yaw of the aircraft (left-and-right motion). Table C.5 summarizes the advantages and disadvantages of two empennage configurations.

Table 5.17: Empennage configuration comparison

<table>
<tr><th></th><th>Fuselage-mounted
</th><th>T-Tail
</th></tr>
<tr><td rowspan="4">Advantages</td><td>The horizontal stabilizer is fixed; if control of the elevator is lost, the aircraft can maintain some pitch.</td><td rowspan="3">Smoother airflow over the elevator. This leads to better pitch control and less drag.</td></tr>
<tr><td>During takeoff (before pitch up), airflow from the engine aids aerodynamics on the elevator.</td></tr>
<tr><td>Simpler design (easier to maintain)</td></tr>
<tr><td>Vertical stabilizer doesn’t have to be as strong (weaker materials are lighter and cheaper)</td><td>Glides farther over a given distance (e.g. if engines fail)</td></tr>
<tr><td rowspan="4">Disadvantages</td><td>Rougher airflow over the elevator. This leads to worse pitch control and more drag.</td><td>More difficult to perform maintenance (difficult to see from the ground, more complex controls)</td></tr>
<tr><td rowspan="3">Doesn’t glide as far over a given distance (e.g. if engines fail)</td><td>Prone to a deep stall condition where the wake of the wing impinges on the tail surface and makes it ineffective.</td></tr>
<tr><td>The entire horizontal stabilizer controls pitch. If the control of the surface is lost, the aircraft cannot control its pitch at all.</td></tr>
<tr><td>Vertical stabilizer has to support the forces generated by the elevator and horizontal stabilizer (stronger materials are heavier or more expensive)</td></tr>
</table>

Rank the categories in terms of what you think are the most important.

1 Safety

2 Maintenance

3 Performance

Discuss the two designs based on your ranking judgement and compare them in a short paragraph.

Table 5.18: Grading scheme for Empennage question

<table>
<tr><th>#</th><th>Quality</th><th>Explanation/Example</th></tr>
<tr><td colspan="3">General response qualities</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Student’s response contains more ideas than those given in the question description.
Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].</td><td>Responses ONLY containing these ideas do NOT have this quality:
• Design complexity as it relates specifically to ease of maintenance
• Airflow over the elevator
• Pitch control once control of flight surfaces is lost
• Glide distance if engines fail
• Strength of materials
• “Deep stall condition”
The student’s response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality.</td></tr>
<tr><td>Example of a response with an “extra” factor:
The maintenance of the fuselage-mounted empennage is easier and allows faster chock-to-chock times. However, it is worse for the movement of airport services on ground.
Other examples include the likelihood of either design catastrophically failing, the implications of each failure for the aircraft, and the applications of each design.</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Student’s response reflects cognitive competencies that successful systems engineers possess.
Justification: Systems engineers have certain cognitive competencies [Frank, 2012].
Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].</td><td>The student’s response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question.
Typically the cookie-cutter, bland student responses do not display this quality. The student’s response must demonstrate that they thought about the problem in a “systems engineering” way.</td></tr>
<tr><td>Understand the whole system and see the big picture: “understand the whole system beyond its elements, sub-systems, assemblies and components, and recognize how each element functions as part of the entire system.”
Example: Things such as glide distance do not matter as much in failure as the majority of flight is at such a high altitude that planes could glide sufficiently far. On the other hand if control is lost the T tail couldn't correct for pitch at all which would be catastrophic</td></tr>
</table>

#	Quality	Explanation/Example
		Understand interconnections: "understand the interconnections and mutual influences and interrelations among system elements." Example: It's largely dependent on the interdependent qualities the design requirements drive. If the aircraft will be performing basic flight profiles for an airline, where maintenance and safety outweigh a small benefit in efficiency, low mounted is better. If the aircraft will be performing high angle-of-attack flight profiles for short airfields with heavy loads, the design may require a T-tail configuration.
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems." Example: Both designs are safe (if designed correctly), though T-tails have some notable safety incidents caused by poor maintenance. These incidents show that maintenance is really a form of safety, and as such I have ranked it with safety.
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level." Example: A difficulty in maintenance can lead to increased costs, schedule delays, or even safety issues if protocol is not followed. The F-M design clearly wins in the maintenance category.
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system." Example: Performance is in the eye of the beholder. Are we looking at performance during normal flight? Performance during an emergency? For normal flight conditions, the T-tail appears to provide better pitch control at a higher risk. The F-M design appears to provide more reliable performance during normal flight. Under emergency situations, some pitch control is maintained with the F-M design whereas the T-tail design loses pitch control at the benefit of a longer glide distance. The comparison here depends on which performance metric is most important.
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	**Able to take into consideration non-engineering factors** Example: As far as maintenance goes, a step ladder would be all that is required to make them equally as easy to inspect which is more of an annoyance than a real detriment.
Question-specific response qualities		

#	Quality	Explanation/Example
4	Student's discussion is consistent with their rankings.	The student's discussion is consistent with their ranking. This could be expressed in various ways: the student mentioned the highest-ranked item first in their discussion, devoted more talking points to it, or explicitly said that the highest-ranked item is the most important for some reason. A response that does NOT have this quality is: [The student ranked "safety" as #1] The T-tall has lower fuel consume, but it cost and maintenance are larger than the Fuselage-Mounted. The student says safety is the most important, but does not discuss it at all or mention why it is the most important.
5	Student's discussion gives justification for their rankings.	The student's discussion describes why they ranked the concepts the way they did. Why is a certain category more important than the other two, or why is a certain category less important than the other two? A response that HAS this quality is: An airline will not buy an aircraft if it is not safe, therefore satisfying safety requirements is the first priority. An airline will also not buy an aircraft if it cannot meet the performance requirements, therefore performance is the second priority. Both Safety and Performance are sufficient conditions. Even if it's a nightmare to maintain, an airline may still buy an aircraft and simply live with the increased operational costs from more maintenance. The converse is not true if the airplane doesn't satisfy the safety and performance requirements. The student discussed why they prioritized one category over another. Note that their claims that an airline will buy an aircraft even if it is a nightmare to maintain are not necessarily correct, and thus do not meet the criteria for quality 8.
6	Student discussed each of the three categories given.	The student's discussion contained remarks on each of the three categories given: maintenance, performance, and safety. They need to explicitly discuss at least one aspect of each of the categories. A response that does NOT have this quality is: The fuselage mounted design would work best since it offers the greatest safety factor and the best option for maintenance. The T tail is more dangerous in the event of lost structure making it unsafe. The student mentioned maintenance and safety, but not performance.

#	Quality	Explanation/Example
7	Student's discussion is accurate and makes sense in the context of the question.	The claims the student makes are accurate and make sense in the context of the question. Based on your experience with aircraft and the information given in the problem, is the student's discussion accurate and does it make sense? Or does the student make some outlandish claims that are totally false? A response that does NOT have this quality is: As far as maintenance goes, a step ladder would be all that is required to make them equally as easy to inspect which is more of an annoyance than a real detriment. This student makes overly-simplified claims in the context of aircraft maintenance.
8	Student's discussion is consistent with their rankings.	The student's discussion is consistent with their ranking. This could be expressed in various ways: the student mentioned the highest-ranked item first in their discussion, devoted more talking points to it, or explicitly said that the highest-ranked item is the most important for some reason. A response that does NOT have this quality is: [The student ranked "safety" as #1] The T-tall has lower fuel consume, but it cost and maintenance are larger than the Fuselage-Mounted. The student says safety is the most important, but does not discuss it at all or mention why it is the most important.

Aircraft Maintenance Question

Your company designs and manufactures aircraft. Your biggest customer for your R-50, a medium-sized jet, uses the aircraft to transport people from Alaska to Mexico. A new customer is considering starting a contract with you for some R-50s, but they want to transport people from Indianapolis to Chicago, a much shorter distance. Both customers have their aircraft in use for 80% of a 24-hour day. How might your maintenance recommendations change for the R-50s for the new customer and why? Consider aircraft aging for 5 aircraft systems, described below.

For the items you recommended changing the maintenance recommendations, please provide a short explanation.

Landing Gear

The landing gear for an aircraft experiences the most severe stresses upon landing.

Engine

The engine of an aircraft experiences wear while operating.

Fuselage/Cabin

The fuselage of an aircraft experiences wear during the pressurization/depressurization cycles of an aircraft.

Landing flaps and spoilers

The landing flaps and spoilers of an aircraft are used to slow the aircraft on final approach to land.

Electronic systems

The electronic systems of an aircraft are used for entertainment, lighting, and control of the aircraft systems.

Sort the following items into the categories.

Items

Landing gear

Engine

Fuselage

Landing flaps and spoilers

Electronics

Should have maintenance recommendations changed for new customer

No change to maintenance recommendations for new customer

For the items you recommended changing the maintenance recommendations, please provide a short explanation.

Table 5.19: Grading scheme for Aircraft Maintenance question

#	Quality	Explanation/Example
General response qualities		
1	Student's response contains more ideas than those given in the question description. Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].	Responses ONLY containing general discussion on these ideas do NOT have this quality: • Landing gear wear is based on number of landings • Engine wear is based on time in operation • Fuselage wear is based on number of flights • Landing flaps and spoilers wear is based on number of landings • Electronic systems wear is based on time in operation The student's response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality.
		Example of a response with an "extra" factor: Since the utilization would remain the same the number of cycles would be drastically increased for the shorter route. This is of course assuming that the maintenance recommendations were based upon hours used and not on number of landings and take-offs. Other examples include ideas like how the aircraft systems are used, what a typical flight profile is like, the regulations surrounding aircraft maintenance, and weather conditions in flight.
2	Student's response reflects cognitive competencies that successful systems engineers possess. Justification: Systems engineers have certain cognitive competencies [Frank, 2012]. Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].	The student's response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question. Typically the cookie-cutter, bland student responses do not display this quality. The student's response must demonstrate that they thought about the problem in a "systems engineering" way.
		Understand the whole system and see the big picture: "understand the whole system beyond its elements, sub-systems, assemblies and components, and recognize how each element functions as part of the entire system." Example: All of the items specifically mentioned increased wear per take-off and landing cycle. Since the utilization would remain the same the number of cycles would be drastically increased for the shorter route. This is of course assuming that the maintenance recommendations were based upon hours used and not on number of landings and take-offs.
		Understand interconnections: "understand the interconnections and mutual influences and interrelations among system elements." Example: Also, the new route will be exclusively in colder climates, so that should be taken into consideration.

#	Quality	Explanation/Example
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems." Example: There will be more cycles as the flights are shorter and the weather conditions will be different.
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level." Example: The landing gear and flaps will be used much more frequently thus needing many more safety checks. The engines will also be used differently, such as short stints rather than long extended trips so the maintenance should reflect that. The fuselage will be experiencing different pressure conditions because the destinations are different so the maintenance should also reflect this change.
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system." Example: The three items, fuselage, landing flaps, and landing gear all will experience significantly more wear by the above listed descriptions. The time of engine use and electronics will likely decrease if they are turned off during the time on the ground (if not, the total time in use is roughly constant.)
		Able to take into consideration non-engineering factors: the student considers non-engineering factors related to the question. Example: You could look into a more efficient engine since the distance is not as far. Also, the environments are very different from Alaska to Mexico vs Indianapolis to Chicago so runways may experience slight changes
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	The grader understands what the student is communicating with minimal confusion or conjecture. This does not necessarily mean that the response is completely spelling- or grammar-error free. Did you understand right away what the student was trying to say?
Question-specific response qualities		
4	Student selected items that will require maintenance changes.	The student sorted each item into two categories: (1) will require maintenance changes, and (2) will not require maintenance changes.
5	Student selected AT LEAST fuselage AND landing gear as wear items that will change.	Since these two systems are based on number of flights, and the new aircraft route will be doing more frequent, shorter flights, these two items must be selected.
6	Student discusses why maintenance changes may occur for each item they selected.	The student discussed why they sorted each item into each category.

#	Quality	Explanation/Example
7	Student's response contains specific ideas on why maintenance changes would occur.	The student's discussion contains specific ideas on why maintenance changes should occur, such as "pressurization/depressurization" or "fatigue" (or related terms). This indicates that they understand how and why the wear occurs on each system. For example, the fuselage/cabin's wear is based on number of flights because for each flight the aircraft pressurizes and depressurizes, which fatigues the metal structure over time and must be checked periodically.
8	Student's specific ideas on why maintenance changes would occur are correct and accurate.	The student's discussion containing specific ideas on why maintenance changes should occur is correct and accurate. An example of an inaccurate/incorrect statement is: Each of these systems will experience more wear due to increased loading cycles, although the duration of the flights / operations will be less on a shorter route.

Vehicle Design Question

You are a project manager for a new off-road-capable day-to-day vehicle at a large car company. A test technician has informed you that the design has just failed the stability test and that it is prone to tipping over completely. He suggests a couple of solutions, and it is your job to figure out which one to decide on. The options are as follows:

1. Add a large plate under the vehicle to increase the weight and make it less prone to tipping. This would reduce the gas mileage.
2. Lower the cabin by replacing the suspension system. This would limit the vehicle's ability to go off-roading.
3. Redesign the vehicle to make it more stable by widening the wheel base. The customer would have to wait longer to buy the vehicle.
4. Change the tires to make the ride smoother and slightly more stable. This doesn't allow the vehicle to drive on as rough terrain.

Rank the options in terms of SAFETY, with 1 being the most safe.

1 Option 1: Large plate
2 Option 2: Shorter suspension
3 Option 3: Redesign
4 Option 4: Reduce tire pressure

Rank the options in terms of COST, with 1 being the cheapest.

1 Option 1: Large plate
2 Option 2: Shorter suspension
3 Option 3: Redesign
4 Option 4: Reduce tire pressure

Rank the options in terms of PERFORMANCE, with 1 being the best performing.

1. Option 1: Large plate
2. Option 2: Shorter suspension
3. Option 3: Redesign
4. Option 4: Reduce tire pressure

Rank the options in terms of TIME TO COMPLETE, with1 being the quickest.

1. Option 1: Large plate
2. Option 2: Shorter suspension
3. Option 3: Redesign
4. Option 4: Reduce tire pressure

Assign a point value to each category that reflects its importance relative to the other categories such that the points add up to 100 (e.g. if category 1 is 20 points and category 2 is 40 points, the assumption is that category 2 is twice as important as category 1).

Safety	0
Monetary Cost	0
Marketability	0
Time to Complete	0
Total	0

Based on your results from the previous questions, which option do you recommend going with? Give a brief explanation why.

Table 5.20: Grading scheme for Vehicle Design question

<table>
<tr><th>#</th><th>Quality</th><th>Explanation/Example</th></tr>
<tr><td colspan="3">General response qualities</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Student’s response contains more ideas than those given in the question description.
Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].</td><td>Responses ONLY containing these ideas do NOT have this quality:
• Gas mileage
• Ability to go off-roading
• Customer wait time
• Driving on rough terrain
The student’s response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality.</td></tr>
<tr><td>Example of a response with an “extra” factor:
If this is a custom vehicle for a certain operator with specific requirements and at an early stage, it would be redesign from scratch. However, if this is for mass markets and just before the mass production, changing of tire pressure and written instruction and/or warning on manual might be the one.
Other examples include the type of customer who would purchase an off-roading vehicle, what the fundamental differences are between off-roading vehicles and everyday vehicles, the market for vehicles, and what the consequences are for vehicles tipping.</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Student’s response reflects cognitive competencies that successful systems engineers possess.
Justification: Systems engineers have certain cognitive competencies [Frank, 2012].
Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].</td><td>The student’s response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question.
Typically the cookie-cutter, bland student responses do not display this quality. The student’s response must demonstrate that they thought about the problem in a “systems engineering” way.</td></tr>
<tr><td>Understand the whole system and see the big picture: “understand the whole system beyond its elements, sub-systems, assemblies and components, and recognize how each element functions as part of the entire system.”
Example: Shortening the suspension takes away some of the off-roading capabilities of the vehicle. I for one would not buy an off-roading vehicle that could not go off road in some places. As a customer, I would prefer to wait for a better designed vehicle that can function and is safe.</td></tr>
</table>

#	Quality	Explanation/Example
		Understand interconnections: "understand the interconnections and mutual influences and interrelations among system elements." Example: Safety is a very serious issue, especially when it comes to potentially rolling over a vehicle, which is not only dangerous to passengers, but effectively compromises the vehicle as well. Taking the time to ensure that a proper solution is implemented could save the company a lot of headache in the long run. It also maintains the selling points of a vehicle, safety, higher mileage, and in this case off-roading capability.
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems." Example: You don't want to give customers false information because that can damage the company's image. Since it is "off-road-capable" car, changing the tire properties cannot be the solution because this would also affect the off road capabilities of the entire vehicle.
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level." Example: The extra weight at the base reduces gas mileage, and car companies are trying to reduce weight as much as possible these days due to environmental regulations.
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system." Example: A heavy plate is not a long term fix, assuming there will be variations of this car model. The company will end up redesigning this issue eventually, might as well do it now.
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	**Able to take into consideration non-engineering factors** Example: I am assuming this is a company that can take a financial loss due to the delay without going bankrupt.
Question-specific response qualities		
4	Student ranked the designs in each category and assigned point values to each category.	The student assigned rankings to each design in each category and assigned point values to each category. If there are any blanks for the point value assignments, or all four categories have the default ranking (i.e. 1, 2, 3, 4), the student's response does NOT have this quality.
5	Student selected an option.	In their discussion, the student selected an option the design team should move forward with in the design. Responses that discuss the "top two" designs are also acceptable.

#	Quality	Explanation/Example
6	Student's ranking and selected option do not contradict each other (+/- one rank).	The grader calculated an "adjusted rank" based on how the student ranked each design option and the point values the student assigned to each category (columns AD through AG have a formula that calculates this and displays it as a rank). For ease of interpretation, the excel sheet has conditional formatting that shows the lowest-ranked design in red and the highest-ranked design in green. If the student's response has met quality 5 and they have selected the best design in their opinion, compare this selection to the "adjusted rank" they gave this (or these) designs. The design they selected should have an adjusted rank of 1 or 2. If the student selected an option that the has an adjusted rank of 4, for example, it would indicate that there is a contradiction between the rankings/point values the student assigned and the design option the student selected and their response does NOT have this quality.
7	Student response contains discussion on why they chose that option by describing why the other 3 design options are worse for any aspect.	The student discussed why they chose a particular design option over the other 3, and they must describe some aspect of all 4 designs (by name) and in some form in their discussion.
8	Student discussed each of the four categories given for the design option they selected.	The student discussed each of the four categories (safety, monetary cost, marketability, and time to complete) for the design option they selected.

Toothbrush Requirement Question

Develop the worst toothbrush imaginable that still satisfies the given requirements. Identify two features of this terrible toothbrush.

Requirement 1: Toothbrush shall last through at least 3 months of use before wear visible to the naked eye occurs for an average user (brushing 2 times per day for 2 minutes with gentle to moderate pressure; 1 to 2).

Requirement 2: Toothbrush shall cost under $0.50 to manufacture.

Requirement 3: Toothbrush shall be chemically nonreactive with common toothpaste formulae (e.g. all toothpaste formulae that the company produces) and water.

Requirement 4: Toothbrush shall be dry to the touch 8 hours exposed in dry air after immersion in water.

First terrible feature:

Second terrible feature:

Requirement:

Table 5.21: Grading scheme for Toothbrush Requirements question

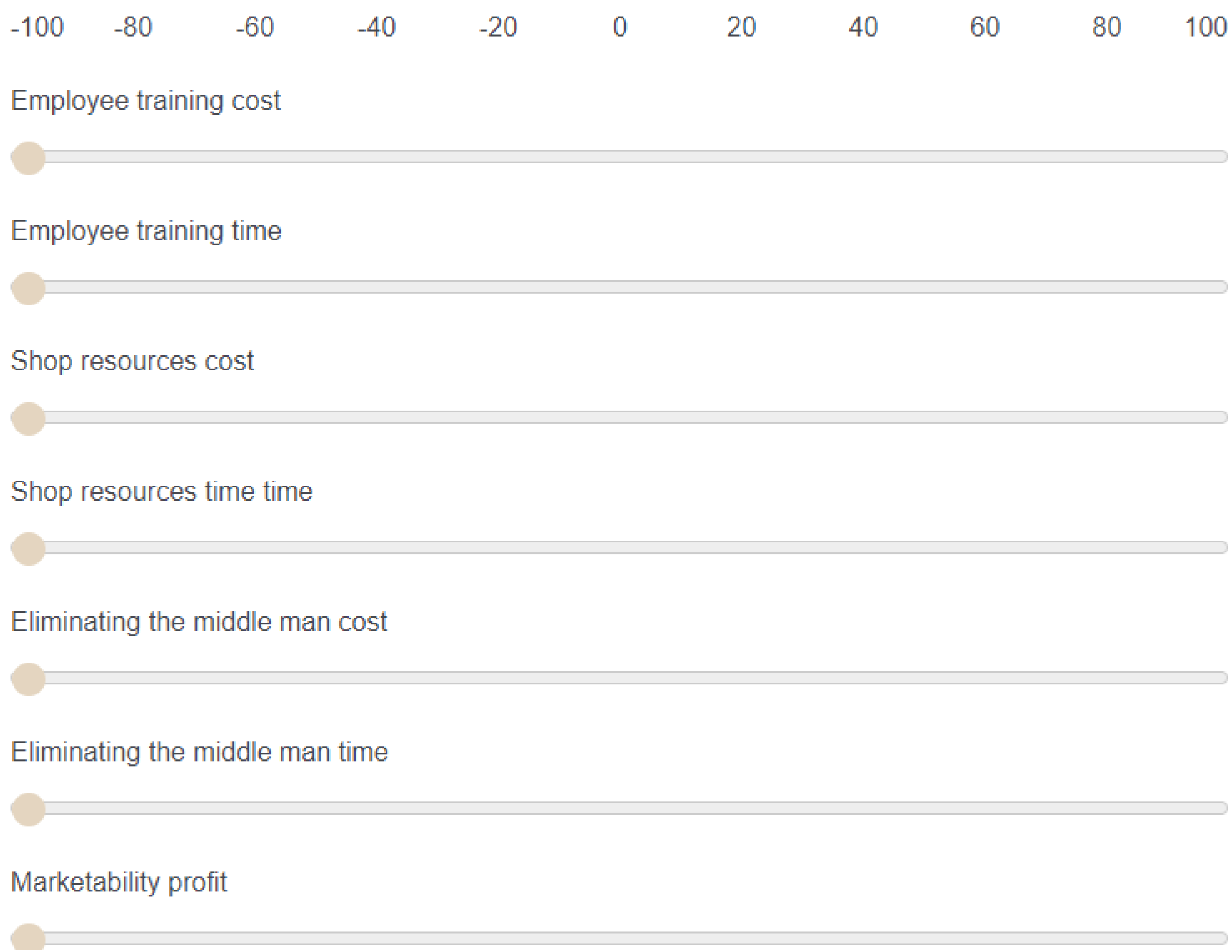
#	Quality	Explanation/Example
General response qualities		
1	Student's response contains more ideas than those given in the question description. Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].	Responses ONLY containing these ideas do NOT have this quality: • Toothbrush lifetime/wear over time • The cost to manufacture the toothbrush • Toothbrush material being chemically nonreactive with toothpaste (other qualities of toothbrush material are OK) • The time it takes the toothbrush to dry after use with water The student's response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality.
		Example of a response with an "extra" factor: (feature) Instead of a stick as a handle, the entire toothbrush is a sphere. this is not efficient to reach the back teeth. Other examples include ideas like toothbrush size, the comfort or safety of the toothbrush as it relates to the user, the weight of the toothbrush, and the shape of the toothbrush.
2	Student's response reflects cognitive competencies that successful systems engineers possess. Justification: Systems engineers have certain cognitive competencies [Frank, 2012]. Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].	The student's response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question. Typically the cookie-cutter, bland student responses do not display this quality. The student's response must demonstrate that they thought about the problem in a "systems engineering" way.
		Understand interconnections: "understand the interconnections and mutual influences and interrelations among system elements." Example: (feature) The handle is a popsicle stick with half of it coated in wax to promote wear-resistance.
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems." Example: (feature) The handle is just round enough, like a cylindrical gomboc, so whenever the user sets it down length-wise, it tips over and the bristles touch the table, but still flat enough that it appears like this shouldn't happen so users do it anyway.
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level." Example: (feature) The tooth brush shall be designed with a removable head to allow easy replacement. (Joint to hold on head is VERY prone to failure while in use)

#	Quality	Explanation/Example
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system." Example: (requirement) Must be demonstrated to reduce plaque and gingivitis with consistent use.
		Able to take into consideration non-engineering factors: the student considers non-engineering factors related to the question. Example: (requirement) The toothbrush must have a designated area designed to clean teeth through direct contact during brushing. The material of this section must be approved by the FDA as being safe for human use (no harmful effects such as toxicity of material or physical damage such as scraping tooth enamel or cutting gums)
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	The grader understands what the student is communicating with minimal confusion or conjecture. This does not necessarily mean that the response is completely spelling- or grammar-error free. Did you understand right away what the student was trying to say?
Question-specific response qualities		
4	Student's response contains two specific "bad" features	The student identified two terrible features of a toothbrush that makes it the "worst toothbrush imaginable". These features are a deviation from the features of a "normal toothbrush", would make the toothbrush overall undesirable to use, and would make people not want to purchase it.
5	Student's "bad" features fits within question-defined requirements	These two terrible features fit within the requirements defined in the question. Example of a feature that would not fit within the requirements: -toothbrush made of gold (>$0.5 to manufacture)
6	Student's response contains an attempt at a requirement	The student attempted to write a requirement, or a design specification, for the toothbrush.
7	Student's requirement negates at least one of their bad features	The student's requirement prevents one of their features from being incorporated into the toothbrush design. Example: (feature) Handle is 3x longer than normal (requirement) Needs to be small/easy to use/transport for normal use and stowing for travel.
8	Student's requirement is measureable	The student's requirement meets the criteria for a "good" requirement, which is that it is verifiable and can be measured. Measurable: toothbrush shall cost less than $0.5 to manufacture. Not measurable: toothbrush shall be cheap to manufacture.

Vehicle Repair Shop Question

You are an owner of an auto repair shop. Your shop does pretty much everything, but you are unsure if you want to start repairing transmissions. Normally you would send them out to a transmission specialty repair shop, but you want to save some money. Consider the impacts of making this change for different aspects of your shop.

Use the sliders to assign each aspect's impact on the shop. A negative value means the aspect is detrimental to the shop in some way, and a positive value means the aspect means the opposite.



What do you think is most important to the shop's success?

1. Employee competance
2. Shop resource allocation
3. Dealing with third parties
4. Marketability

Do you think offering transmission repair is worth it?

Table 5.22: Grading scheme for Vehicle Repair Shop question

#	Quality	Explanation/Example
General response qualities		
1	Student's response contains more ideas than those given in the question description. Justification: Systems Engineers are creative [Frank, 2012; Hutchison & Verma, 2018; Derro & Williams, 2009].	Responses ONLY containing general discussion on these ideas do NOT have this quality: • Employee training time • Shop resources • Eliminating the "middle man" (also "dealing with third parties") • Marketability The student's response must contain some novel idea that indicates they thought about the problem and did not simply regurgitate information from the question. Think about what you would expect students to discuss when faced with this question, and anything that surprises you meets this quality.
		Example of a response with an "extra" factor: Yes, but only if they do enough transmission repairs to justify the cost to bring everything in-house. If they only do a couple per quarter then it doesn't make sense, but if they do two per week it would probably be cost effective to bring it in-house. Other examples include ideas like whether other shops in the area do transmission repairs, the company's relationship with their transmission repair shop and their customers, the demand for transmission repairs in the area, and how the additional skills may influence their employees.
2	Student's response reflects cognitive competencies that successful systems engineers possess. Justification: Systems engineers have certain cognitive competencies [Frank, 2012]. Systems engineers have a systems mindset and have the ability to do big-picture thinking [Hutchison & Verma, 2018].	The student's response reflects the following cognitive competencies of successful systems engineers from Frank [2012] (the most applicable subset of the 16 factors) as they apply to the question. Typically the cookie-cutter, bland student responses do not display this quality. The student's response must demonstrate that they thought about the problem in a "systems engineering" way.
		Understand the whole system and see the big picture: "understand the whole system beyond its elements, sub-systems, assemblies and components, and recognize how each element functions as part of the entire system." Example: If a capable technician is available or already on staff as to avoid initial training costs and the shop had sufficient capital to invest in the tooling and parts they need to have on hand for most repairs. I think it could be worth it to do a very limited expansion without trying to make it a pillar of the business. The value added would be on limiting the amount of outside work required and not increasing the prestige of the shop. A larger expansion would be possible after more techs were trained but wouldn't be a realistic starting goal.

#	Quality	Explanation/Example
		Understand interconnections: "understand the interconnections and mutual influences and interrelations among system elements." Example: If the employees can be trained in a timely manner (as not to cease all ongoing work in the shop while they are trained), then it absolutely is. A more widely-knowledgeable shop that has more capabilities can better cater to its customers than a one-trick shop.
		Understand system synergy (emergent properties): "able to identify the synergy and emergent properties of combined systems." Example: I think it would be worth it if the shop can absorb the upfront cost without huge detriment, but it's reward would accumulate over time.
		Understand the system from multiple perspectives: "avoid adopting a one-dimensional view and are able to describe a system from all relevant perspectives that go beyond the mere engineering level." Example: No. At this point, transmissions are constantly evolving from 6-speed to 7,8,9. There are also dual clutch and single clutch versions of all these transmissions. There are also CVT and not all cvt are equal. I would simply replace the transmission with a used one.
		Understand the implications of proposed change: "able to analyze the impact of proposed changes and are capable of anticipating and dealing with all implications of changes in the system." Example: Yes, I think offering transmission repair is worth it since the shop offers auto repairs anyway and would have a lot of resources and equipment on hand that would be required to perform transmission repair.
		Able to take into consideration non-engineering factors: the student considers non-engineering factors related to the question. Example: It depends on how easily trained the employees are, and how intensive the training would be
3	Student's response is clearly communicated. Justification: "Communication is critical for systems engineers since they interact with a variety of people" [Hutchison & Verma, 2018].	The grader understands what the student is communicating with minimal confusion or conjecture. This does not necessarily mean that the response is completely spelling- or grammar-error free. Did you understand right away what the student was trying to say?
Question-specific response qualities		
4	Student gave a value to each aspect and that value matches their decision.	The student assigned an importance value to each aspect. The grader should calculate the overall cost and time by multiplying the values the student assigned to each category and the ranking they gave to each category. If the overall cost and time are severely negative, and the student said the shop should offer repairs, this indicates an inconsistency.

#	Quality	Explanation/Example
5	Student indicated whether the shop should provide transmission repairs.	The student made a decision on whether or not the shop should provide transmission repairs. They said “yes” or “no”.
6	Student discussed why the shop should or should not provide transmission repairs.	The student discussed why the shop should or should not provide transmission repairs.
7	Student discussed each of the four categories given.	The student’s discussion contains ideas from each of the four categories (employee training, shop resources, eliminating the middle man, and marketability).
8	Student’s discussion contains specific ideas or comparisons between pros and cons for at least one of the categories.	The student’s discussion contains specific ideas or comparisons between pros and cons for at least one of the categories.

APPENDIX D. CARTS DIAGRAMS

This section contains the CARTs diagrams discussed in Section 3.4.2.

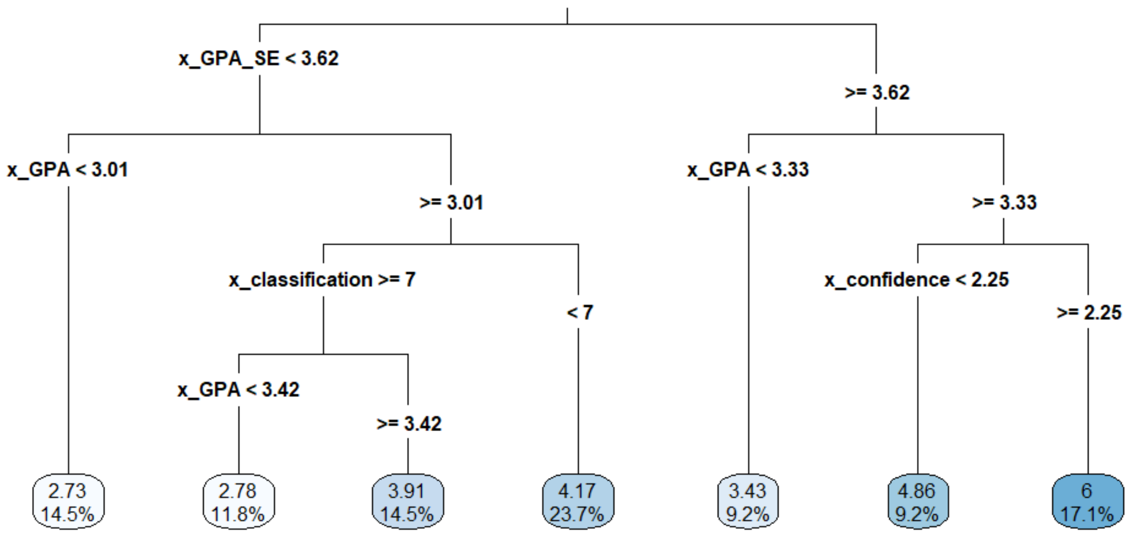


Figure 5.8: Undergraduate decision tree for Levee question

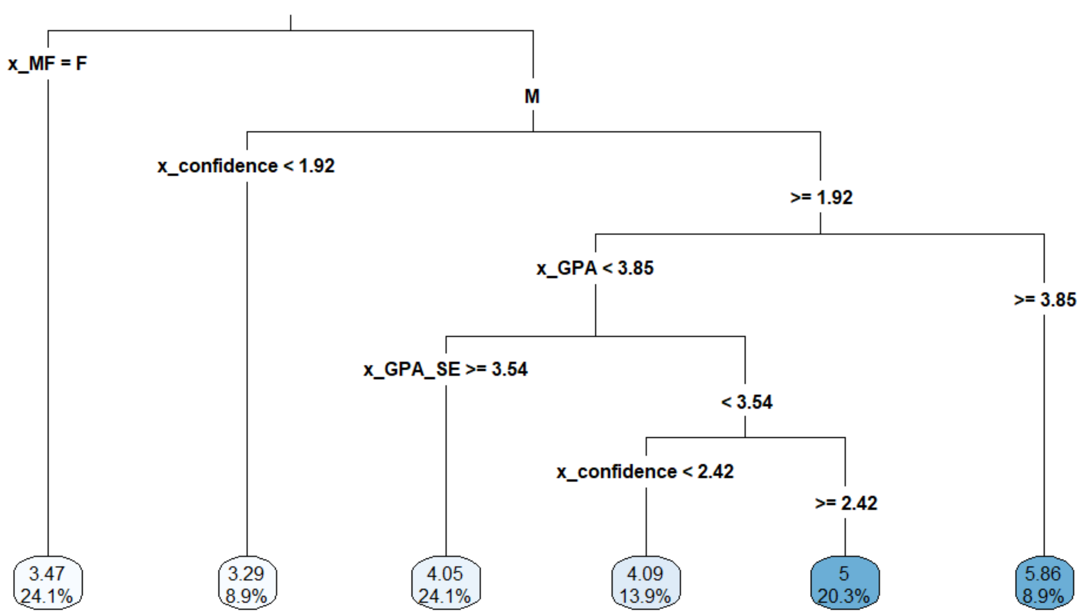


Figure 5.9: Undergraduate decision tree for Boat Race question

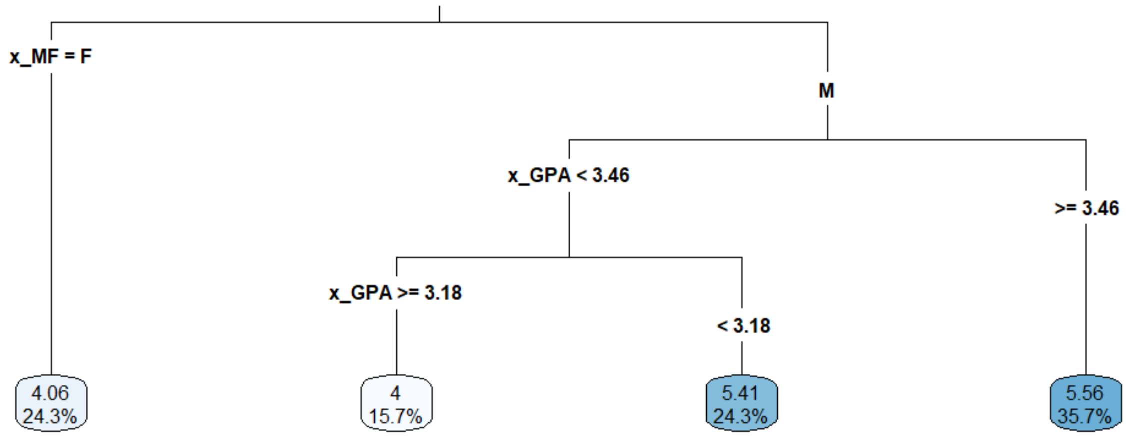


Figure 5.10: Undergraduate decision tree for Empennage question

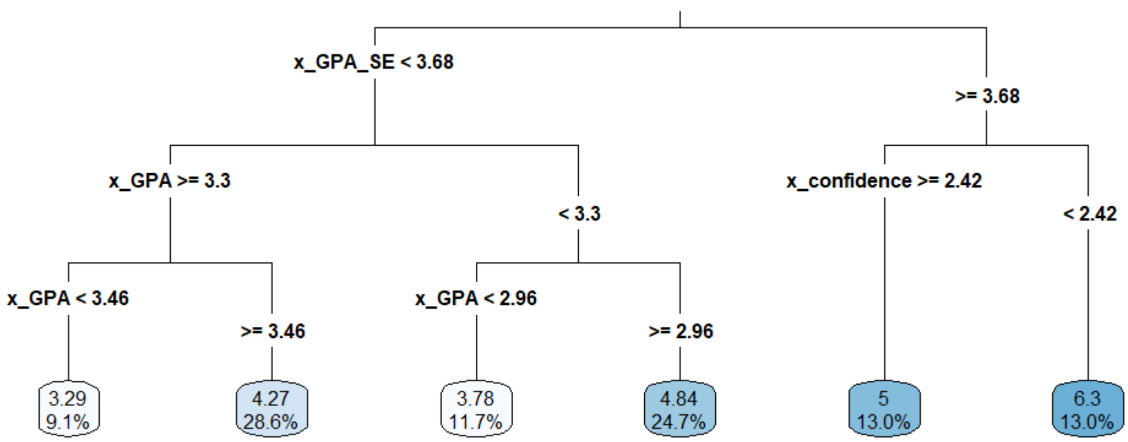


Figure 5.11: Undergraduate decision tree for AC Maintenance question

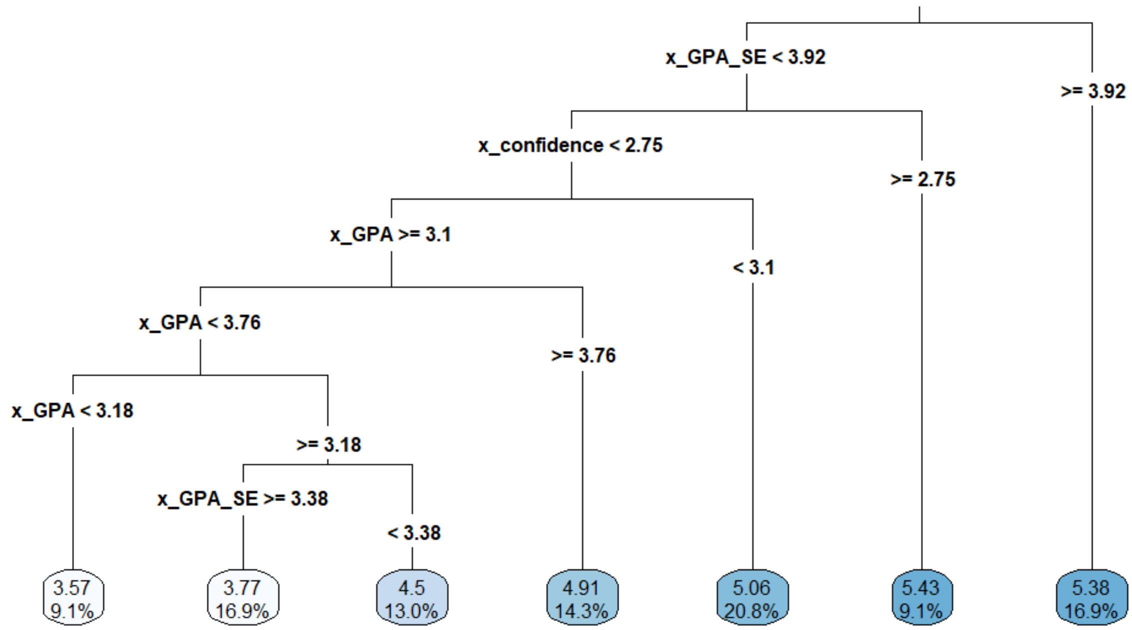


Figure 5.12: Undergraduate decision tree for Car Design question

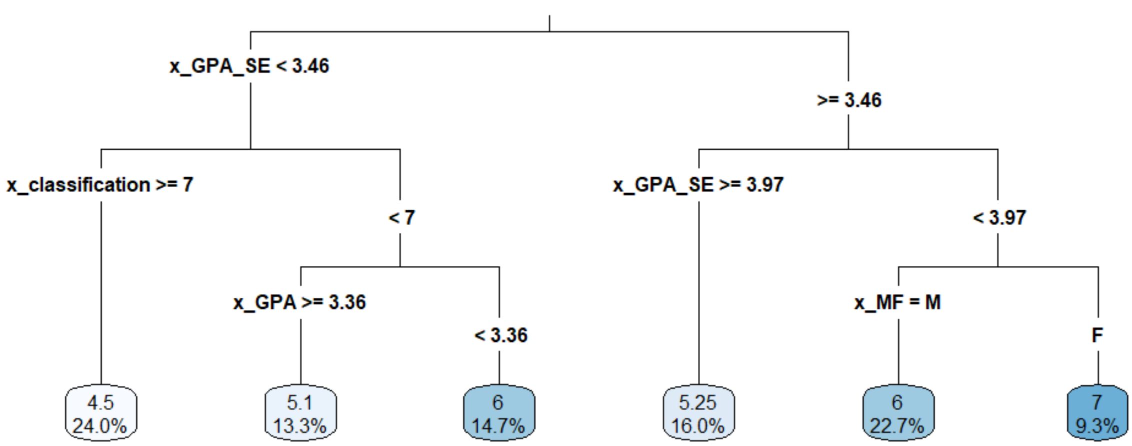


Figure 5.13: Undergraduate decision tree for Toothbrush question

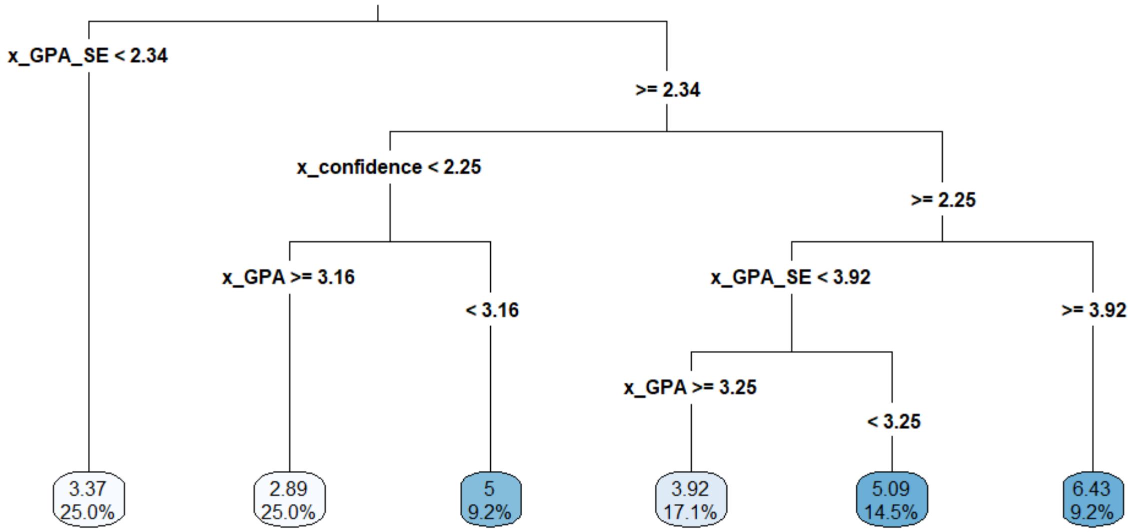


Figure 5.14: Undergraduate decision tree for Transmission question

APPENDIX E. USABILITY SURVEY

Have you used this network before?

Your project has problems with communication. Try to find advice on that problem.

	Yes	No
Were you able to find advice?		
Was this advice difficult to find?		
Was this advice easy to understand?		
Was this advice potentially useful?		

What do you want to click on first?

What do you think will happen when you click on that?

Click on it. What happened? Did you expect that?

Any other input from your first impression?

Any general input on the look and feel of the network?

Your project has problems with communication. Try to find advice on that problem.

	Yes	No
Were you able to find advice?		
Was this advice difficult to find?		
Was this advice easy to understand?		
Was this advice potentially useful?		

Your project has problems with management. Try to find advice on that problem.

	Yes	No
Were you able to find advice?	○	○
Was this advice difficult to find?	○	○
Was this advice easy to understand?	○	○
Was this advice potentially useful?	○	○

You think that people in management roles may have more problems than other types of employees. Try to find problems specifically relating to managers.

	Yes	No
Were you able to find advice?	○	○
Was this advice difficult to find?	○	○
Was this advice easy to understand?	○	○
Was this advice potentially useful?	○	○

Your project is specifically in the gas and oil industry. Try to find problems specifically in a similar industry.

	Yes	No
Were you able to find advice?	○	○
Was this advice difficult to find?	○	○
Was this advice easy to understand?	○	○
Was this advice potentially useful?	○	○

Any additional thoughts on using the network?

How can the network be improved (for instance by adding functionality)?

APPENDIX F. USEFULNESS SURVEYS

Helios Aircraft Mishap

Figure 5.15: Helios aircraft [NASA, 2004]

Summary: The NASA-developed Helios aircraft was a solar- and fuel-cell powered, unmanned, flying-wing aircraft designed for long-term, high altitude flights for service as an atmospheric satellite, atmosphere data collector, and communication platform. During a test flight, the aircraft encountered turbulence, which caused its structure to flex too far and break up during flight [NASA, 2004].

Table F.1 contains the five findings we selected to present to the industry representatives from the Helios aircraft mishap.

Table 5.23: Helios aircraft mishap report findings

#	Finding from report	Finding summary
1	“Two flight hazards were identified and considered relevant to the longitudinal instability experienced on the mishap flight […] Though several causes were listed, the first hazard report failed to identify the specific mechanism leading to an unstable and quickly diverging phugoid response, as well as, the interactive effects of many factors that might contribute to realization of such an outcome. The probability of this hazard was assessed in its mitigated state as “remote”. The second hazard report recognized the possibility that weather could directly exceed design limitations and was assessed as more likely than an instability and divergent control response. Absent from all hazard reports was any evidence of the probability of occurrence that supported both the unmitigated and mitigated probability of occurrence for the hazards.” [NASA 2004, p. 79]	The Helios team identified flight hazards that could have led to flight instability or structural damage, but in their hazard reports they did not identify the mechanisms leading to these hazards or give supporting evidence for the low probabilities they assigned to these hazards.
2	“It was determined that adding three point masses to the HP01 [Helios Prototype flown in 2001] configuration contributed to causing the persistent high dihedral. Although the Helios aircraft was conceived as a very simple aircraft design for high altitude solar flight, the structural flexibility and the large masses associated with the fuel cell system introduced substantial complexity into the aircraft’s flight dynamics. [...] For the HP03 [Helios Prototype flown in 2003] configuration at normal speeds and no turbulence, the wing dihedral varied on an average from about 11 feet to about 17 feet tip deflection during the first flight on June 7, 2003. It is concluded that the persistent high dihedral caused the pitch instability.” [NASA 2004, p. 67]	The original aircraft design was relatively simple, but changes to the aircraft's fuel system made the fuel components heavy. These components were mounted to the aircraft structure in a way that caused the structure to flex more than initially anticipated during flight, even at normal speeds and with no air turbulence.
3	“[T]he tools and the solution techniques were deemed to be inadequate for predicting the vehicle’s sensitivity to disturbances and lack of robustness to return to a low dihedral condition. Additionally, validating math models and predictions using ground and flight test data was found to be inadequate.” [NASA 2004, p. 78]	The design and verification of the aircraft were inadequate because the analysis models and tools the team used were inadequate. For example, the turbulence model the team used did not capture non-linear flight dynamics behavior. Additionally, the models the team used were not verified using ground and flight test data, despite the team recognizing uncertainties in the predictions.

#	Finding from report	Finding summary
4	"The pilot's primary display was designed to involve the pilot in selecting autopilot choices for navigation, the airspeed, the power setting, and FCS gains; it was not designed to allow direct control of aircraft attitude through elevator or power control. [...] the pilot was afforded visual cues through cameras that provided selectable views in orthogonal directions. Since part of the pilot's workload was to monitor the control system stability, the use of visual cues was crucial in early recognition of instabilities, however monitoring of dihedral and pitch stability were at odds to each other since each required a different camera view which was not possible simultaneously. Furthermore, the fidelity of the forward view suffered from poor horizon definition as a result of picture quality, haze, and the lack of terrain features for the test route. Typically, a trained pilot uses the horizon (or artificial horizon) for important cues for initiating recover from an aircraft departure." [NASA 2004, p. 85]	The pilot did not recognize that the aircraft flight dynamics were becoming unstable because the pilot's primary display was inadequate. The pilot was not able to monitor both pitch stability and structural flex using the same camera view, despite being responsible for monitoring both.
5	"The crew failed to recognize the ever-increasing instability. This was exacerbated by the previous missions success and encounters that afforded long times for recognition and response. Performance predictions had not indicated such stability issues. For this reason, the crew was pre-conditioned by a benign first flight and subsequently surprised by the in-flight departure. Secondly, criteria and ill-defined crew roles and responsibilities concerning instability recognition compounded the problem. The lack of response to the pilot's initial inquiry concerning the pitch response was indicative of this problem. In addition, a photo helicopter witnessed the event and was manned by an aerospace engineer who was very familiar with the vehicle, but was not linked to a command and control frequency." [NASA 2004, p. 84]	The Helios crew did not recognize that the aircraft flight dynamics were becoming unstable. This was partially because there were no clear roles and responsibilities if instability was recognized: non-pilot crew members were present, but were not explicitly told to give input on the flight. Additionally, the crew was overconfident that the flight would succeed because of past flight success.

Helmet Water Intrusion

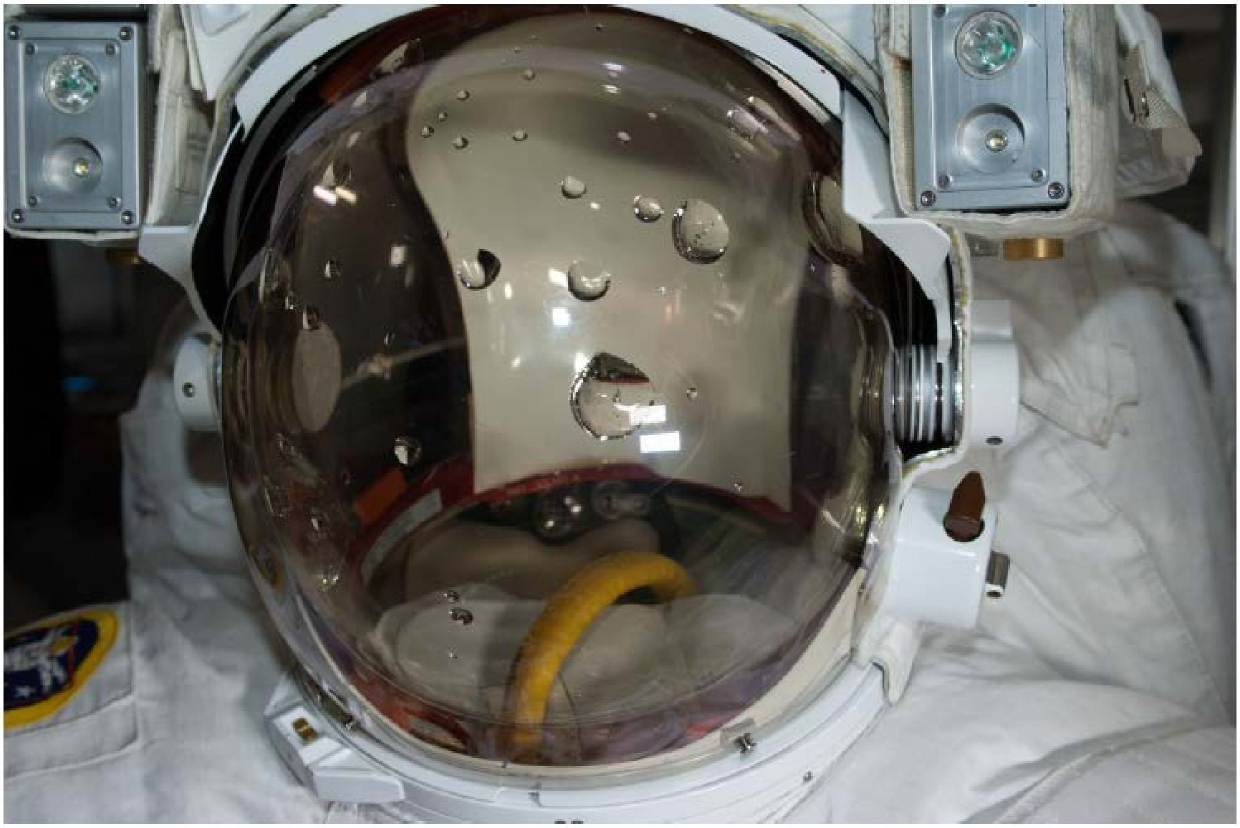


Figure 5.16: EVA suit water intrusion picture [NASA, 2013]

Summary: A crewmember of the ISS experienced water filling his helmet while performing Extravehicular Activity (EVA), starting at the back of his head and eventually migrating onto his face. The crewmember experienced impaired visibility and breathing, as well as audio communication issues due to the water and terminated the EVA early. The crewmember was able to safely make it back onto the ISS and remove the helmet without experiencing injury [NASA, 2013].

Table F.2 contains the five findings we selected to present to the industry representatives from the EVA suit water intrusion incident.

Table 5.24: Helmet water intrusion mishap report findings

#	Finding from report	Finding summary
1	"After EVA 22, the team perceived that lengthy meetings were not possible, because of the high ops tempo involved in preparing for EVA 23. Essentially, the Flight Control Team accepted the crew's assessment of the EVA 22 water leak and chose not to investigate further. According to post mishap interview transcripts, more than one team member indicated that they wished they had called a "time-out." However, EVA 23 was scheduled for the following week, which left little time to prepare. There was also a perception that if the question concerning the source not being the drink bag was raised, it would invoke a fairly resource intensive and potentially cumbersome process involving Engineering and Safety for what most felt would likely turn out to be a non-issue. This would have an impact on EVA 23 preparations. In hindsight, however, it is now apparent that EVA 23 should not have commenced until the EVA 22 issue had undergone a more adequate evaluation. That is not to say that a lengthy formal risk assessment was required (that may, or may not be the case), just that the EVA 22 water leak deserved a more refined assessment of risk. Had that been done, the EVA 23 HVCC might not have occurred." [NASA 2013, p. 84]	The ground team for the ISS program believed that there was no time to investigate a previous incident involving water intrusion in a helmet during an EVA, so they decided to commence with the next one without evaluating what had happened. The ground team wanted to avoid initiating a lengthy formal risk assessment process and thus put the next EVA at risk.
2	"Through interviews with ground personnel and review of data from previous EMU [Extravehicular Mobility Unit] performance, it was clear that some water entering the helmet was considered normal by the ground teams. Despite the fact that water carryover into the helmet presented a known hazard of creating eye irritation due to its interaction with anti-fog agents, and also presented a potential fogging hazard, the ground teams grew to accept this as normal EMU behavior. Since these smaller amounts of water carryover had never caused a significant close call, it was perceived to not be a hazardous condition. When water began entering EV2's helmet, the ground team discussed anti-fog/eye irritation concerns and visibility concerns; however, a more hazardous condition was not expected because the presence of water in the helmet had been normalized. [NASA 2013, pp. 15-16]	The ground crew considered water entering the helmet a normal event, despite the significant risks it posed to the crewmember wearing the suit. Similar events on a smaller scale had occurred in the past without incident and the team thus did not consider that a more hazardous situation could occur.

#	Finding from report	Finding summary
3	“Through interviews and review of flight rules and procedures, the MIB learned that while there is a significant amount of knowledge about the way water behaves in zero-gravity, the ground teams did not properly understand how the physics of water behavior inside the complex environment of the EMU helmet would manifest itself. Engineering teams had 1-g experience showing that a significant amount of water in the vent loop would stall the vent fan. However, during this HVCC, an amount of water that would normally stall the fan during ground testing was allowed to pass by the fan and enter the helmet. The Engineering teams now believe that, in zero-g, the water can cling to the interior walls of the fan housing and be passed through the fan assembly without stalling it, which they presume happened during this HVCC. The MIB concurs with this explanation. In addition, the Engineering teams informed the MIB that they believed that if a significant amount of water entered the helmet, the air flow from the vent loop would force the water to streak over the top of the helmet and down the front of the visor, possibly affecting the crew member’s visibility. From evaluation of this event, the MIB and ground teams now know that the water entered the helmet, but did not streak over the top and down the visor. Instead, surface tension forced the water to form near the outlet of the vent line until the quantity was sufficient enough to contact the back of EV2’s head. At that point, surface tension brought the large amount of water to the back of EV2’s head and it eventually made its way to the front of EV2’s head, covering his eyes and nostrils.” [NASA 2013, pp. 94-95]	The ground team did not understand how water would behave in the crewmember’s helmet. The team believed that water would cling to the inside of the helmet and would thus alert the crewmember to the presence of the water and that the most significant risk would be to impair the crewmember’s visibility. However, surface tension kept the water at a specific location in the suit and the crewmember was not alerted to the presence of water until a significant amount had collected and the crewmember was in danger of asphyxiation. Had the ground team applied knowledge of the physics of water behavior in zero-g, the team may have better anticipated the risks associated with water in the EVA suit.
4	“Based on interviews and comm loop recordings, it was found that the ground team (Engineering, Safety, Operations) primarily focused on EV2’s drink bag as the possible source of water in his helmet. Other suggestions included accumulation of sweat and leakage from the LCVG, but both were quickly dismissed. Channelizing on the drink bag may have prevented the team from continuing to ask questions to come up with a different answer or ask new and more specific questions that would have pointed to something other than the drink bag, such as the temperature of the water. When the CO2 sensor failed early in the EVA at GMT 12:35, most of the team believed that it failed due to a nominal accumulation of moisture in the vent loop. Since nominal water carryover only results in a limited/manageable amount of water in the helmet, the significance of the CO2 sensor failure was quickly disregarded, despite the fact that this type of failure almost always occurred near the end of a long EVA. No one on the team recognized the relationship between the early failure of the CO2 sensor and an abnormally large amount of water in the vent loop until much later.” [NASA 2013, pp. 89-90]	The ground team mistakenly contributed the water in the helmet to the suit’s drink bag and dismissed other potential sources, despite evidence that a bigger problem had occurred. They thus did not investigate other potential causes.

#	Finding from report	Finding summary
5	"From interviews and discussions with personnel, it was determined that updating the FMEA is primarily viewed by many as a paperwork exercise and a tool to be mainly used by the S&MA [Safety & Mission Assurance] community. This is further evidenced in practice by the lack of time and effort taken to update and review the information when it is deemed necessary to update as well as its lack of involvement in engineering risk discussions or training." [NASA 2013, p. 96]	The ground team was not adequately using the tools it had available to assess risk on the program's activities. The team viewed updating and reviewing the FMEA as a paperwork exercise and subsequently did not use this tool when discussing risk on the program or training.

“With Tool” Scenario Questions

For each finding:

1. Given this limited information, what remediation measure(s) would you suggest for NASA’s finding? How would you propose solving this problem?
2. On a scale of 1 to 10, how useful would your remediation be at alleviating NASA’s finding? Do you think your remediation is something that would actually work? 10 is most useful, 1 least useful.
   If the representative has suggested more than one remediation measure:
3. On a scale of 1 to 10, how useful would your remediation be at alleviating NASA’s finding? Do you think your remediation is something that would actually work? 10 is most useful, 1 least useful.
4. On a scale of 1 to 10, how useful would your remediation be at alleviating NASA’s finding? Do you think your remediation is something that would actually work? 10 is most useful, 1 least useful.

After going through all of the findings and showing the participant the tool:

1. On a scale of 1 to 10, how useful is the tool overall? 10 is most useful, 1 least useful.
2. Do you think this tool could be useful in your own work context?
3. Would you consider using it at your company?
4. What improvements to the tool would you suggest?
5. Any other input

“Without Tool” Scenario Questions

For each finding:

1. What remediation measures would you suggest to alleviate NASA’s finding, using what you found in the tool?
2. On a scale of 1 to 10, how useful would your remediation be at alleviating NASA’s finding? Do you think this is something that would actually work? 10 is most useful, 1 least useful.
3. On a scale of 1 to 10, how helpful was the tool to develop this remediation measure? 10 is most helpful, 1 least helpful.
   If the representative has suggested more than one remediation measure:
4. On a scale of 1 to 10, how useful would your remediation be at alleviating NASA’s finding? Do you think this is something that would actually work? 10 is most useful, 1 least useful.
5. On a scale of 1 to 10, how helpful was the tool to develop this remediation measure? 10 is most helpful, 1 least helpful.

After going through all of the findings:

6. On a scale of 1 to 10, how useful is the tool overall? 10 is most useful, 1 least useful.
7. Do you think this tool could be useful in your own work context?
8. Would you consider using it at your company?
9. What improvements to the tool would you suggest?
10. Any other input?

APPENDIX G. ADDITIONAL FEEDBACK ON THE TOOL

At the conclusion of each interview, we asked each interviewee to interact with the tool and answer general questions on aspects like its usability and potential improvements we could make, regardless of whether they had used the tool to answer questions on the NASA findings. Thus, all 21 systems engineers we interviewed provided general feedback on the tool. Table G.1 displays the additional feedback we received from each interviewee on the tool that may form the basis for updates we make to the tool.

Table 5.25: Additional feedback on the tool from each interviewee

Description	Feedback
Expert 01 CB	Use this tool as a teaching tool in a classroom environment. Not solely at academic institutions, but also for training at companies for new systems engineers. Link the tool to INCOSE standards (or other systems engineering standard processes); they have some processes to follow when you're INCOSE certified. Did you have a risk plan? Did you follow it? Did you use a risk board [i.e. a board of people who review risks on a project]? Did you have a requirements validation [asking specific questions about the project]? Being a systems engineer means something different at each specific aerospace company, so standardizing this tool even across organizations in the same disciplines may be difficult. If you make the tool industry-wide [i.e., apply systems engineering standards to the tool], nuclear and aerospace for example are very similar. But if you try to apply to other industries, it will get more and more difficult.
Expert 02 CD	He would use the tool if it was populated with their own company-specific data [like a lessons-learned database]. They have volumes of their own data and go through it rigorously.
Expert 03 DA	The tool did not help with developing the design [e.g., early in the design cycle], but it helped identify and address potential issues. It's in a unique category of tools: identify potential pitfalls and potholes. There may not be many tools out there that are doing this. It may be valuable. Multiple cause selections would be useful. Some findings had multiple types of causes that would be useful to see together in the tool. Provide the tool platform as a blank template that a company could use to disseminate information on their own internal failures Show the gap between what the accident investigators recommended and what the company actually implemented based on the recommendations, and the recommendations' impact on safety or performance.

Description	Feedback
Expert 04 DC	If we want to automate the process, it may be easy to extract the data but will be a different story for writing the narratives. Semantic analysis would help make the narratives searchable, and help automate the narrative writing process (and maybe help make the narratives more uniform if multiple people are writing them). Automating the failure addition process would be useful [look into information on Triz, or tech optimizer]: we have already looked into doing machine learning for failure report analysis, but how could we automate the story addition? Use semantic analysis on the stories to analyze them and build a database to make future stories, which he had encountered in a systems engineering/project management class he had taken. This representative had vision impairment [he said text size, not contrast, was the problem], allowing the user to “minimize” the random story so they can zoom in on the network and still view all of it at once.
Novice 01 CB	Give the user an opportunity to input information from their own project. Different insight could be useful (i.e. FMEA paperwork criticisms). He was concerned that the body investigating the accidents would not criticize their own processes. If you play around with the tool you can get through a lot of the information but a tutorial video or instruction manual may be useful. It also may scare some people away if they don’t want to read a 100-page instruction manual. Develop some way of introducing the user to the tool without scaring them. An opinion of an engineer versus a manager could be interesting.
Novice 02 CD	The links between causes aren’t currently filtered by “who was involved in the failure”. He filtered the causes down by “personnel (operations)”, then clicked on “inadequately communicated” (which had stories under Buncefield). This cause had a link to “managed risk poorly”, which did not have a corresponding story under Buncefield because all of the “managed risk poorly” stories for this accident were for operations management. Consider putting the “causes” and “recommendations” next to the network instead of the random story. He thinks it’s more important and useful to have that list next to the network.
Novice 03 DA	Font size could be a problem, but the zoom-in feature worked ok.
Novice 04 DC	Can the tool have a web-md kind of functionality where you put in your symptoms and see potential consequences or what the causes may be? It would need to be a bit more interactive.
Novice 05 CD	It would be good if there was a search function. If the user could input “any recommendation about engines” for example would help the user whittle the tool down to their own context.
Novice 06 CB	Have the list of recommendations reduce down to what is connected to the cause you clicked on, maybe in a different tab up at the top.
Novice 07 DC	Link the random story to its place in the list. Also, potentially provide a link to more information on the failure (the user could just google, but that’s another step), potentially to the Wikipedia page of the failure. It was also unclear how severe the failure was, as in how many people died. The significance of line weight is not very obvious. There is a disconnect because the cause/recommendation in the story doesn’t have a link to the network. It wasn’t clear that the alphabetical list of causes/recommendations was interactive. The student thought they would have to search through the network to find the right node to get more information.
Novice 08 DA	He should be able to add his own recommendations into the tool easily. Then he could click on all the things the recommendation could be applicable to and the network consumes it.

Description	Feedback
Novice 09 DC	Consider introducing a new feature: selecting multiple causes at once to see the recommendations they have in common. Also consider showing the specific corresponding map for each accident when you click on “view stories”. The lines connecting nodes make it difficult to read the node names, consider making them a bit lighter. It really matters to see the line weight when you click on a node to see its connections. Consider having two versions of the tool: one with no lines and the other with all the connections.
Novice 10 CB	It was a little difficult to read at first, consider increasing the font size. There could be a search function where a user could input a keyword. “Physics based failure”. Once the database becomes more populated a search query would be very helpful. Consider using some form of picture representation, like a tree. If you make a certain set of decisions, what could be the outcome? If you pick two causes to improve upon, for example, what could happen?
Novice 11 DA	It would be nice to see specific causes and recommendations related to a single story. Click on the “Alaska airlines” accident and see how its causes and recommendations are connected specifically. It currently does grouping for accident type (e.g. aerospace) but not for specific accidents.
Novice 12 CD	Consider changing node size or font size to make it more obvious which nodes were connected more frequently.
Novice 13 DC	Allow the user to select more than one node to see for example what recommendations two causes have in common. Show a list of the causes, and allow the user to check the ones that apply to them. Then show a list of recommendations, and how many times they applied to that checked list of causes. The user could start with the recommendations that applied the most to the list of causes they specified. Reflect this checked list of causes on the network and show the connections with recommendations. The user could specify how important each cause is to them to help define the links.
Novice 14 CB	Each cause has multiple options and could cause the engineer to overthink the problem. It may cause someone to become “addicted” to the tool and won’t help them focus in on a good solution. When she reads a finding, she has an idea of how many causes could potentially apply to it. Then when she uses the network, she sees recommendations common to all and is more likely to look into those recommendations more. Offering too much help vs. no help at all is a delicate process because it could squash creativity. Many of the systems people were not interested in reading the stories because they were inspired by the network itself and the connections. Maybe consider including some sort of help that shows them they can refer to the stories below to get more ideas/details.
Novice 15 CD	Include some sort of search bar. He would search for “design failure” or something more specific in the accident stories. “wing failure” or “propeller failure”. Adding more cases would be helpful for this because then there would be more specific things to search for. Consider adding key words for each accident. (Alaska: maintenance, lubrication, horizontal stabilizer; Aloha: corrosion)

Description	Feedback
Novice 16 DA	Consider displaying some weight on the nodes to indicate how prevalent they are in the data. Consider displaying the data in different ways—with a histogram or pie chart perhaps? The tool is good because it helps to narrow recommendations down to important ideas. It helped him organize his thoughts and come up with recommendations. It helped him focus on what could be the most important recommendations.
Novice 17 CB	A back button would be useful to see your previous selections.

REFERENCES

Abedin, Muhammad Arshad Ul, Vincent Ng and Latifur Khan. "Cause Identification from Aviation Safety Incident Reports via Weakly Supervised Semantic Lexicon Construction." Journal of Artificial Intelligence Research 38 (2010): 569-631.

Action, James M. and Hibbs, Mark. Why Fukushima Was Preventable. Washington, D.C.: Carnegie Endowment for International Peace, 2012.

Adcock, Rick, et al. "Systems Engineering Education for All Engineers." Rome: IEEE International Symposium on Systems Engineering, 2015.

Aloisio, D. C. and K. Marais. "Preventing slips, overruns, and cancellations: Application of accident investigations and theory to the understanding and prevention of engineering project failures." West Lafayette: Purdue University Research Repository, 2017. <https://purr.purdue.edu/publications/2859/1>.

—. "Preventing slips, overruns, and cancellations: Application of accident investigations and theory to the understanding and prevention of engineering project failures." (2017).

Aloisio, Diane C, Karen Marais and Hanxi Sun. "Development of a Survey Instrument to Evaluate Student Systems Engineering Ability." ASEE Annual Conference & Exposition (2018).

Alonso-Saldivar, Ricardo. Probe Finds Reasons for Obamacare Website Failure. 31 July 2014. 31 March 2015. <www.inc.com/associated-press/management-failure-results-in-healthcare.gov-woes.html>.

Armstrong, James R. and Jon Wade. "Development of Systems Engineering Expertise." Procedia Computer Science 44 (2015): 689-698.

ASC. "In-Flight Breakup Over the Taiwan Strait China Airlines flight CI611." 2002.

ASCE. The New Orleans Hurricane Protection System: What Went Wrong and Why. Reston: American Society of Civil Engineers Hurricane Katrina External Review Panel, 2007.

Bagg, Thomas C. III, et al. "Systems Engineering Education Development (SEED) Case Study." INCOSE Thirteenth Annual International Symposium (2003).

Baker, James, et al. The Report of The BP U.S. Refineries Independent Safety Review Panel. Texas City: BP, 2007.

Barstow, David, David Rohde and Stephanie Saul. "Deepwater Horizon's Final Hours." The New York Times 25 December 2010.

Bar-Yam, Yaneer. When Systems Engineering Fails --- Toward Complex Systems Engineering. Washington D.C.: IEEE International Conference on Systems, Man and Cybernetics, 2003.

BEA. Final Report. Le Bourget Cedex: Bureau d’Enquêtes et d’Analysespour la sécurité de l’aviation civile, 2016.

Ben-Gal, I. "Bayesian Networks." Encyclopedia of Statistics in Quality & Reliability. Wiley & Sons, 2007.

Bergin, Chris. "X-33/VentureStar - What really happened." 4 January 2006. NASASpaceflight.com. <www.nasaspaceflight.com/2006/01/x-33venturestar-what-really-happened/>.

Bilalić, Merim et al. "Mechanisms and Neural Basis of Object and Pattern Recognition: A Study With Chess Experts." Journal of Experimental Psychology 139 (2010): 728-742.

Björklund, Tua A. "Initial mental representations of design problems: Differences between experts and novices." Design Studies 34 (2013): 135-160.

Bradsher, Keith. "Risky Decision/A Special Report.; Study of Ford Explorer's Design Reveals a Series of Compromises." The New York Times 7 December 2000.

Bylin, Kyle. "4 Likely Causes of Micosoft Zune Death." 16 March 2011. hypebot. <www.hypebot.com/hypebot/2011/03/4-likely-causes-of-the-death-of-the-zune.html>.

Chanda, Sasanka Sekhar and Sougata Ray. "Learning from Project Failure: Globalization Lessons for an MNC." Wiley Online Library (2015).

Chandrasekaran, Rajiv. "F-35's ability to evade budget cuts illustrates challenge of paring defense spending." The Washington Post 9 March 2013.

Chaput, Armand J. and Mark, Hans M. "Systems Engineering Design - An Educational Imperitive for Development of the Future Aerospace and Defense Workforce." Los Angeles: Aviation Technology, Integration, and Operations Conference, 2013.

Chaput, Armand J. "Issues in Undergraduate Aerospace System Engineering Design Education - An Outsider View from Within." Fort Worth: AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, 2010.

—. "System Engineering Education for All Engineers - A Capstone Design Approach." New Orleans: ASEE's 123rd Annual Conference & Exposition, 2016.

Charette, Robert. "What's Wrong with Weapons Acquisitions?" IEEE Spectrum 1 November 2008.

Ciralsky, Adam. "Will it Fly?" Vanity Fair 16 September 2013. February 2014. <http://www.vanityfair.com/politics/2013/09/joint-strike-fighter-lockheed-martin>.

Claybrook, Joan. "On Firestone Tire Defect and Ford Explorer Rollovers." Washington, D.C.: United States Senate Committee on Commerce, Science and Transportation, 2000.

Conrady, Stefan and Lionel Jouffe. Bayesian Networks & BayesiaLab. Franklin: Bayesia USA, 2015.

Cooke, David L. "A system dynamics analysis of the Westray mine disaster." System Dynamics Review 19.2 (2003): 139-166.

Coolman, Ashley. "Lessons Learned from Project Failure at Denver International Airport: Why Checking Bags is Still a Pain." 30 October 2014. Wrike. <https://www.wrike.com/blog/lessons-learned-from-project-failure-at-denver-international-airport-why-checking-bags-is-still-a-pain/>.

Cowing, Keith. "Iridium: If You Build It They Won't Necessarily Come." 29 August 2000. spaceref. <www.spaceref.com/news/viewnews.html?id=208>.

Cross, Nigel. "Expertise in Design: an overview." Design Studies 25.5 (2004): 427-441.

CSB. Explosion and Fire at the Macondo Well. Washington, D.C.: United States Chemical Safety and Hazard Investigation Board, 2014.

—. Final Report: Kleen Energy Urgent Recommendations. Washington, D.C.: United States Chemical Safety and Hazard Investigation Board, 2010.

—. Refinery Explosion and Fire. Washington, D.C.: United States Chemical Safety and Hazard Investigation Board, 2007.

—. Sugar Dust Explosion and Fire. Washington, D.C.: United States Chemical Safety and Hazard Investigation Board, 2008.

—. Xcel Energy Hydroelectric Plant Penstock Fire. Washington, D.C.: United States Chemical Safety and Hazard Investigation Board, 2007.

CTI. Hearing on "How to Improve the Efficiency, Safety and Security of Maritime Transportation: Better use and Integration of Maritime Domain Awareness Data". Washington, D.C.: United States Committee on Transportation and Infrastructure, 2013.

CTSA. "Inquiry into the fire on Heavy goods Vehicle shuttle 7539 on 18 November 1996." 1997.

Cullen, William. The Public Inquiry into the Piper Alpha Disaster. London: United Kingdom Department of Energy, 1990.

Davidz, Heidi L., Deborah J. Nightingale and Donna H. Rhodes. Enablers and Barriers to Systems Thinking Development: Results of a Qualitative and Quantitative Study. Hoboken: 3rd Conference on Systems Engineering Research, 2005.

de Neufville, Richard. "The Baggage System at Denver: Prospects and Lessons." Journal of Air Transport Management 1.4 (1994): 229-236.

DefenseNews. "Top 100 for 2018." 2 October 2018. <https://people.defensenews.com/top-100/>.

Denning, Steve. "What Went Wrong At Boeing?" 21 January 2013. Forbes. March 2014. <http://www.forbes.com/sites/stevedenning/2013/01/21/what-went-wrong-at-boeing/>.

Derro, Mary Ellen and Christine R. Williams. "Behavioral competencies of highly regarded systems engineers at NASA." Aerospace conference 2009 IEEE (2009): 1-17.

Dilger, Daniel Eran. "Lessons from the Death of HD-DVD." 21 February 2008. RoughlyDrafted. <www.roughlydrafted.com/2008/02/21/lessons-from-the-death-of-hd-dvd/>.

DOT. Management Advisory on Weaknesses With Site-Specific Deployment Requirements and Specialist Training for STARS. Washington, D.C.: United States Department of Transportation, 2014.

Doukakis, J.P. "Exploration of Expert and Novice Reasoning in Mechanics of Solids." Salt Lake City: ASEE Annual Conference & Exposition, 2018.

Dumais, Susan T. "Latent Semantic Analysis." American Society for Information Science and Technology 38.1 (2004): 188-230.

Dvorak, John C. "Vista's 11 Pillars of Failure." 21 April 2008. PCMag. <www.pcmag.com/article2/0,2817,2286065,00.asp>.

Eckerman, Ingrid. The Bhopal Saga--Causes and Consequences of the World's Largest Industrial Disaster. Bhopal: Universities Press (India), 2005.

Enderle, Rob. "RIP Zune: How Microsoft could have succeeded." 15 March 2011. TG Daily. <www.tgdaily.com/opinion/54687-rip-zune-how-microsoft-could-have-succeeded>.

ESA & NASA. SOHO Mission Interruption. Greenbelt: Joint ESA/NASA Investigation Board, 1998.

Farand, Patrice and Jason R. Tavares. "A concept inventory for knowledge base evaluation and continuous curriculum improvement." Education for Chemical Engineers 21 (2017): 33-39.

Felder, Richard M., et al. "A Longitudinal Study of Engineering Student Performance and Retention. III. Gender Differences in Student Performance and Attitudes." Journal of Engineering Education 84.2 (1995): 151-163.

Finkelstein, Sydney and Shade Sanford. "Learning from Corporate Mistakes: The Rise and Fall of Iridium." Organizational Dynamics 29.2 (2000): 138-148.

Frank, Moti. "Engineering Systems Thinking: Cognitive Competencies of Successful Systems Engineers." Procedia Computer Science 8 (2012): 273-278.

Gallagher, Sean. "Littoral failure: Navy hedges bets on high-tech littoral combat ships." 28 March 2014. Ars Technica. <arstechnica.com/tech-policy/2014/03/littoral-failure-navy-hedges-bets-on-high-tech-littoral-combats/>.

GAO. Defense Acquisitions; Assessments of Selected Weapon Programs. Washington D.C.: United States Government Accountability Office, 2017.

—. Navy Ships: Lessons or Prior Programs May Reduce New Attack Submarine cost Increases and Delays. Washington, D.C.: United States General Accounting Office, 1994.

—. Problems Continue to Plague the Seawolf Submarine Program. Washington, D.C.: United States General Accounting Office, 1993.

Gehman, Harold, et al. Columbia Accident Investigation Board Report Volume I. Washington, D.C.: United States National Aeronautics and Space Administration, 2003.

Gelbart, Marcia. "New radar system cleared to take off STARS, which was being tested by air traffic controllers in Philadelphia, won U.S. approval. The cost: $1 billion-plus." 10 June 2003. philly. <articles.philly.com/2003-06-10/business/25447572_1_radar-system-terminal-automation-replacement-system-air-traffic-control-system>.

Gelinas, Nicole. "Lessons of Boston's Big Dig." Autumn 2007. City Journal. <www.city-journal.org/html/17_4_big_dig.html>.

Gertler, Jeremiah. "V-22 Osprey Tilt-Rotor Aircraft: Background and Issues for Congress." 2009.

Ghaoui, Laurent El, et al. "Understanding Large Text Corpora via Sparse Machine Learning." Statistical Analysis and Data Mining 6.3 (2013): 221-242.

Graham, Bob and William Reilly. "Deep Water: The Gulf Oil Disaster and the Future of Offshore Drilling." 2011.

Hallgren, Kevin A. "Computing Inter-Rater Reliability for Observational Data: An Overview and Tutorial." Tutor Quant Methods Psychol. 8.1 (2012): 23-34.

Henderson, Jerald M., et al. "Building the Confidence of Women Engineering Students With a New Course to Increase Understanding of Physical Devices." Journal of Engineering Education 83.4 (1994): 337-342.

Hollmer, Mark. "Segway poses challenge for pedestrians and insurers." Insurance Times XXI.17 (2002).

Honan, Mat. "Remembering the Apple Newton's prophetic failure and lasting impact." 5 August 2013. Wired. <www.wired.com/2013/08/remembering-the-apple-newtons-prophetic-failure-and-lasting-ideals/>.

Hormby, Tom. "The Story Behind Apple's Newton." 6 August 2013. Low End Mac. <lowendmac.com/2013/the-story-behind-apples-newton/>.

Howell, Kellan. "FAA's new air traffic control system unstable, less capable than old system." 18 August 2014. Washington Times. <www.washingtontimes.com/news/2014/aug/18/faas-terminals-system-upgrade-would-waste-438m-ig/?page=all>.

Hutchison, Nicole A.C. and Dinesh Verma. An Update to the Theory of Effective Systems Engineers. Hoboken: Systems Engineering Research Center, 2018.

Hutchison, Nicole, Devanandham Henry and Art Pyster. "Atlas: Understanding What Makes Systems Engineers Effective in the U.S. Defense Community." Systems Engineering 19 (2016).

IAEA. The Chernobyl Accident. Vienna: International Atomic Energy Agency, 1992.

Ingram, Carl. "DMV Spent $$ Million on Failed Project." 27 April 1994. LA Times. <articles.latimes.com/1994-04-27/news/mn-50941_1_modernization-project>.

Jackson, Scott and Timothy L. J. Ferris. "Resilience principles for engineered systems." Systems Engineering 16.2 (2013): 152-164.

James, Erika H. "Decision Making under Pressure." Chicago: CFA Institute, 2015.

James, Gareth, et al. "Tree-Based Methods." An Introduction to Statistical Learning with Applications in R. Ed. Casella, G., Fienberg, S. and Olkin, I. New York: Springer, 2017. 303-335.

Johnson, Kirk. "Denver Airport Saw the Future. It Didn't Work." 27 August 2005. The New York Times. <http://www.nytimes.com/2005/08/27/us/denver-airport-saw-the-future-it-didnt-work.html>.

JPL Special Review Board. Report on the Loss of the Mars Polar Lander and Deep Space 2 Missions. Pasadena: United States National Aeronautics and Space Administration Jet Propulsion Lab, 2000.

Jugdev, Kam and Ralf Müller. "A Retrospective Look at Our Evolving Understanding of Project Success." Project Management Journal 36.4 (2005): 19.

Kaeding, Nicole. "The Failed HealthCare.gov Launch." Cato Institute 26 January 2015.

Keil, Mark and Magnus Mähring. "Is Your Project Turning into a Black Hole?" California Management Review 53.No. 1 (2010).

Kemeny, John G., et al. The President's Commission On The Accident at Three Mile Island. Washington, D.C.: The President's Commission on the Accident at Three Mile Island, 1979.

Kennedy, Edward M. and John Jr. Conyers. Navy Ships: Lessons of Prior Programs May Reduce New Attack Submarine Cost Increases and Delays. Washington, D.C.: United States General Accounting Office, 1994.

Khan, Faisal I. and Amyotte, Paul R. "Modeling of BP Texas City refinery incident." Journal of Loss Prevention in the Process Industries 20 (2007): 387-395.

Kim, Jieun and Hokyoung Ryu. "A Design Thinking Rationality Framework: Framing and Solving Design Problems in Early Concept Generation." Human-Computer Interaction 29 (2014): 516-553.

Konstandinidou, Myrto, et al. "In-depth analysis of the causal factors of incidents reported in the Greek petrochemical industry." Reliability Engineering and System Safety 96 (2011): 1448-1455.

Krug, Steve. Rocket Surgery Made Easy: The Do-It-Yourself Guide to Finding and Fixing Usability Problems. First. Berkeley: New Riders, 2009.

Kumar, Anita. "Attention shifts from Firestone to Ford Explorer." 17 June 2001. St. Petersburg Times. <www.sptimes.com/News/061701/news_pf/Firestone/Attention_shifts_from.shtml>.

Kurokawa, Kiyoshi. "The official report of The Fukushima Nuclear Accident Independent Investigation Commission." 2012.

Laing, Robert D. Report of the Commission of Inquiry into matters relating to the safety of the public drinking water in the City of. Regina: The office of the Queen's Printer, 2002.

Langewiesche, William. "The Lessons of ValuJet 592." March 1998. The Atlantic. <https://www.theatlantic.com/magazine/archive/1998/03/the-lessons-of-valujet-592/306534/>.

Leveson, Nancy. "A new accident model for engineering safer systems." Safety Science 42 (2004): 237-270.

—. Evaluating Accident Models using Recent Aerospace Accidents. Boston: Software Engineering Research Laboratory; Aeronautics and Astronautics Dept.; Massachusetts Institute of Technology, 2001.

—. Safeware: System Safety and Computers. Addison-Wesley Professional, 1995.

—. Systems safety engineering: Back to the future. Boston: MIT Aeronautics and Astronautics, 2002.

Lineberger, Robin S. and Aijaz Hussain. Program management in aerospace and defense; Still late and over budget. New York: Deloitte Development LLC, 2016.

Lipton, Eric. "Security Effort by Coast Guard Is Falling Short." 30 December 2006. The New York Times. <www.nytimes.com/2006/12/30/us/30domain.html?pagewanted=print>.

Lockheed Martin. "F-35 Program Timeline." 2018. 24 8 2018. <https://www.f35.com/about/history>.

MacDonald, Gordon. "Buncefield: Why did it happen?" 2011.

Marais, Karen, Joseph Saleh and Nancy Leveson. "Archetypes for organizational safety." Safety Science 44 (2006): 565-582.

Marks, Gene. "How Google Screwed Up Google Glass." 21 April 2014. Forbes. <www.forbes.com/sites/quickerbettertech/2014/04/21/how-google-screwed-up-google-glass/>.

Mayer, Kenneth R. The Development of the Advanced Medium-Range Air-to-Air Missile: A Case Study of Risk and Reward in Weapon System Acquisition. Santa Monica: RAND Corporation, 1993.

McAteer, J Davitt. Upper Big Branch: The April 5, 2010 explosion: a failure of basic coal mine safety practices. Charleston: Governor's Independent Investigation Panel, 2011.

McCracken, Harry. "Sixteen Reasons the Windows Vista Era Never Quite Happened." 10 August 2009. Technologizer. <www.technologizer.com/2009/08/10/sixteen-reasons-the-windows-vista-era-never-quite-happened/>.

McCullagh, Peter. "The proportional-odds model." 29 May 2013. September 2017. <http://www.stat.uchicago.edu/~pmcc/reports/prop_odds.pdf>.

Mcintosh, Bill. "Down to earth reasons for Iridium failure." 19 August 1999. The Independent. <www.independent.co.uk/news/business/down-to-earth-reasons-for-iridium-failure-1113638.html>.

Mellow, Craig. "The Rise and Fall and Rise of Iridium." September 2004. Air & Space Magazine. <https://www.airspacemag.com/space/the-rise-and-fall-and-rise-of-iridium-5615034/?all>.

Mishap Investigation Board. Mars Climate Orbiter. Washington, D.C.: United States National Aeronautics and Space Administration, 1999.

Mokey, Nick. "Five Reasons the Microsoft Zune is Dead." 18 May 2009. Digital Trends. <www.digitaltrends.com/mobile/five-reasons-the-microsoft-zune-is-dead/>.

Morgan, David and Cornwell, Susan. "Contractors describe scant pre-launch testing of U.S. healthcare site." 24 October 2013. Reuters. <www.reuters.com/article/2013/10/24/us-usa-healthcare-idUSBRE99M0VD20131024>.

Mouawad, Jad. "Report on Boeing 787 Dreamliner Battery Flaws Finds Lapses at Multiple Points." 1 December 2014. The New York Times. <www.nytimes.com/2014/12/02/business/report-on-boeing-787-dreamliner-batteries-assigns-some-blame-for-flaws.html>.

Myerson, Allen R. "Automation Off course in Denver." 18 March 1994. The New York Times. <http://www.nytimes.com/1994/03/18/business/automation-off-course-in-denver.html?pagewanted=all>.

NASA. Aviation Safety Reporting System Program Briefing. Washington, D.C.: United States National Aeronautics and Space Administration, 2015. <https://asrs.arc.nasa.gov/overview/summary.html>.

—. International Space Station (ISS) EVA Suit Water Intrusion High Visibility Close Call. Washington, D.C.: United States National Aeronautics and Space Administration, 2013.

—. Investigation of the Helios Prototype Aircraft Mishap. Washington, D.C.: United States National Aeronautics and Space Administration, 2004.

—. The Hubble Space Telescope Optical Systems Failure Report. Washington, D.C.: United States National Aeronautics and Space Administration, 1990.

NDIA. Report on Systemic Root Cause Analysis of Program Failures. Arlington: U.S. National Defense Industrial Association Systems Engineering Division, 2008.

Neapolitan, Richard E. Learning Bayesian Networks. Upper Salle River: Prentice-Hall, 2003.

Nelson, Brian. "Top 10 Things People Hate About Vista." 6 January 2009. pluralsight. <https://www.pluralsight.com/blog/it-ops/top-10-reasons-people-hate-windows-vista>.

Newman, JS. "Failure-Space: A Systems Engineering Look at 50 Space System Failures." Acta Astronautica 48.5-12 (2001): 517-527.

Novet, Jordan and Sullivan, Mark. "Feds say agile development contributed to Healthcare.gov failure." 30 July 2014. Venturebeat. <venturebeat.com/2014/07/30/u-s-govt-heres-what-went-wrong-with-healthcare-gov/>.

Nowinski, Edmund H. and Robert J. Kohler. "The Lost art of Program Management in the Intelligence Community." 15 April 2007. Central Intelligence Agency. 1 June 2018. <https://www.cia.gov/library/center-for-the-study-of-intelligence/csi-publications/csi-studies/studies/vol50no2/html_files/Program_Management_4.htm>.

NRC. Report on the Accident at the Chernobyl Nuclear Power Station. Washington, D.C.: United States Nuclear Regulatory Commission, 1987.

NTSB. Aloha Airlines, Flight 243. Washington, D.C.: United States National Transportation Safety Board, 1989.

—. Aviation Coding Manual. Washington, D.C.: United States National Transportation Safety Board, 1998.

—. Grounding of the U.S. Tankship Exxon Valdez on Bligh Reef Near Valdez, Alaska. Washington, D.C.: United States National Transportation Safety Board, 1989.

—. In-flight Breakup Over the Atlantic Ocean Trans World Airlines flight 800 DCA96MA070. Washington, D.C.: United States National Transportation Safety Board, 1996.

—. In-flight Fire and Impact with Terrain ValuJet Airlines Flight 592. Washington, D.C.: United States National Transportation Safety Board, 1996.

—. Loss of control and Impact with Pacific Ocean Alaska Airlines Flight 261. Washington, D.C.: United States National Transportation Safety Board, 2000.

—. Loss of Control on Approach Colgan Air, Inc. Operating as Continental Connection Flight 3407 DCA09MA027. Washington, D.C.: United States National Transportation Safety Board, 2009.

—. "Safety Board Determines Cause of Boston's Big Dig Tunnel Ceiling Collapse Last Year." 10 July 2007. United States National Transportation Safety Board. <https://www.ntsb.gov/news/press-releases/Pages/Safety_Board_Determines_Cause_of_Bostons_Big_Dig_Tunnel_Ceiling_Collapse_Last_Year.aspx>.

Nutt, Paul C. Why Decisions Fail: Avoiding the Blunders and Traps that Lead to Debacles. San Francisco: Berrett-Koehler Publishers, Inc., 2002.

Oberg, James. "Why the Mars Probe went off course." 1 December 1999. IEEE Spectrum. <spectrum.ieee.org/aerospace/robotic-exploration/why-the-mars-probe-went-off-course>.

O'Connor, Dennis. Report of the Walkerton Inquiry. Toronto: Ontario Ministry of the Attorney General, 2002.

OIG. Audit of the Drug Enforcement Administration's Aviation Operations with the Department of Defense in Afghanistan. Washington, D.C.: United States Office of the Inspector General, 2016.

—. Audit Report: Review of the V-22 Aircraft Program. Washington, D.C.: United States Office of the Inspector General, 1994.

—. Audit Report: V-22 Osprety Joint Advanced Vertical Aircraft. Washington, D.C.: United States Office of the Inspector General, 2000.

—. F-35 Lightning II Program Quality Assurance and Corrective Action Evaluation. Washington, D.C.: United States Office of the Inspector General, 2015.

—. Quality Assurance Assessment of the F-35 Lightning II Program. Washington, D.C.: United States Office of the Inspector General, 2013.

Pachal, Pete. "Why Google Glass Isn't the Segway." 2 April 2015. Mashable. <mashable.com/2013/07/29/why-google-glass-isnt-the-segway/>.

Panckhurst, Graham. Report of the Royal Commission on the Pike River Coal Mine Tragedy. Wellington: Royal Commission on the Pike River Coal Mine Tragedy, 2012.

Paté-Cornell, Elisabeth. "Learning from the Piper Alpha Accident: A Postmortem Analysis of Technical and Organizational Factors." Risk Analysis 13.2 (1993).

Pavlovich, J. Gregory and Rea-Dix, Charlotte L. . Formal Report of Investigation of the 30 April 1999 Titan IV B/Centaur TC-14/Milstar-3 (B-32) Space Launch Mishap. Washington, D.C.: United States Air Force Accident Investigation Board, 1999.

Peduzzi, Peter, et al. "A Simulation Study of the Number of Events per Variable in Logistic Regression Analysis." Journal of clinical Epidemiology 49.12 (1996): 1373-1379.

Pernin, Christopher G. Lessons from the Army's Future Combat Systems Program. Santa Monica: RAND Corporation, 2012.

Pinto, J K and S J Mantel. "The Causes of Project Failure." IEEE Transactions on Engineering Management 37.4 (1990): 269-276.

Pinto, Jeffrey K. and Dennis P. Slevin. "Critical Factors in Successful Project Implementation." IEEE Transactions on Engineering Management EM-34.1 (1987).

Prabhakaran, Selva. "R-Statistics." 2016. 15 August 2018. <http://r-statistics.co/Linear-Regression.html>.

Rasmussen, Jens. "Risk Management in a Dynamic Society: A Modelling Problem." Safety Science 27.2/3 (1997): 183-213.

Reason, J. "The contribution of latent human failures to the breakdown of complex systems." Philosophical Transactions of the Royal Society of London Series B, Biological Sciences 327 (1990): 475-484.

Richard, Peter. Report of the Westray Mine Public Inquiry. Halifax: Queen's Printer for Nova Scotia, 1997.

Rivlin, Gary. "Segway's Breakdown." Wired 1 March 2003.

Robinson, S. D., W. J. Irwin and X. O. Wu. "Application of machine learning to mapping primary causal factors in self reported safety narratives." Safety Science 75 (2015): 118-129.

Rogers, William P. Report of the Presidential Commission on the Space Shuttle Challenger Accident. Washington, D.C.: Presidential Commission on the Space Shuttle Challenger Accident, 1986.

Ryschkewitsch, Michael, Dawn Schaible and Wiley Larson. The Art and Science of Systems Engineering. Washington, D.C.: U.S. National Aeronautics and Space Administration, 2009.

Saleh, J.H., et al. "Highlights from the literature on accident causation and system safety: Review of major ideas, recent contributions, and challenges." Reliability Engineering and System Safety 95 (2010): 1105-1116.

Schachtman, Noah. "Pentagon Chief Rips Heart Out of Army's 'Future'." 6 April 2009. Wired. <www.wired.com/2009/04/gates-rips-hear/>.

Schneid, Joe. "Wiki Commons." 2 July 2006. <https://upload.wikimedia.org/wikipedia/commons/d/d4/EllstromManufacturingHydroplane.jpg>.

SERC. Helix--Phases 1 and 2. Hoboken: Systems Engineering Research Center, 2013.

Shappell, Scott A. and Douglas A. Wiegmann. The Human Factors Analysis and Classification System--HFACS. U.S. Federal Aviation Administration. Washington, D.C.: United States Department of Transportation, 2000.

Shore, Barry. "Systematic Biases and Culture in Project Failures." Project Management Journal 39.4 (2008): 5-16.

Shweta, Bajpai, R.C. and Chaturvedi, H.K. "Evaluation of Inter-Rater Agreement and Inter-Rater Reliability for Observational Data: An Oveview of Concepts and Methods." Journal of the Indian Academy of Applied Psychology 41.3 (2015): 20-27.

Skinner, Samuel and William Reilly. The EXXON VALDEZ Oil Spill. Washington, D.C.: The National Response Team, 1989.

Slegers, Nathan J., et al. "Learning from failure in systems engineering: A panel discussion." Systems Engineering 15.1 (2011): 74-82.

Smith, Albert E. "OpEd: Where's the Leadership?" 2 November 2007. Space News. 1 June 2018. <http://spacenews.com/oped-wheres-leadership/>.

Smith, Michael W. Boston's "Big Dig": A Socio-Historical and Political Analysis of Malfeasance and Official Deviance. Manchester: National Social Science Association, 2010.

Smith, R. Jeffrey. "Premier U.S. Fighter Jet Has Major Shortcomings." 10 July 2009. The Washington Post. <http://www.washingtonpost.com/wpdyn/>.

Smith, Will. "Exclusive Interview: Microsoft Admits What Went Wrong with Vista, and How They Fixed It." 10 September 2008. MaximumPC. <www.maximumpc.com/article/features/exclusive_interview_microsoft_admits_what_went_wrong_with_vista_and_how_they_fixed_it>.

Solomon, Brian. "How Tableau Built A $3 Billion Data Empire On Top Of Beautiful Charts." Forbes 31 May 2016.

Sorenson, Diane C. "Accidental knowledge: Using accidents and other project failures to inform research in systems engineering." 2015.

Squires, Alice F., et al. "Investigating an Innovative Approach for Developing Systems Engineering Curriculum: The Systems Engineering Experience Accelerator." Vancouver: ASEE Annual Conference & Exposition, 2011.

Stricklin, Kevin G. Upper Big Branch Accident Investigation Report. Washington, D.C.: United States Department of Labor, 2010.

Svedung, I. and J. Rasmussen. "Graphic representation of accident scenarios: mapping system structure and the causation of accidents." Safety Science 40 (2002): 397-417.

Takahashi, Dean. "Xbox 360 defects: an inside history if Microsoft's video game console woes." GabesBeat 5 September 2008. 27 Febrary 2015. <venturebeat.com/2008/09/05/xbox-360-defects-an-inside-history-of-microsofts-video-game-console-woes/view-all/>.

Taubman, Philip. "Costly Lesson on How Not to Build a Navy Ship." The New York Times 25 April 2008. "Failure to Launch: In Death of Spy Satellite Program, Lofty Plans and Unrealistic Bids." The New York Times 11 November 2007.

Texas A&M University. "Colleges & Departments." 8 October 2018. <https://www.tamu.edu/about/departments.html>.

Thamhain, Hans. "Managing Risks in Complex Projects." Project Management Journal 44.2 (2013): 20-35.

The Boeing Company. "F-15 Eagle Tactical Fighter Historical Snapshot." 2018. 24 08 2018. <https://www.boeing.com/history/products/f-15-eagle.page>.

The Pennsylvania State University. "The Two-Stage Nested Design." 2018. 27 September 2018. <https://onlinecourses.science.psu.edu/stat503/node/69/>.

Thomas, Allister. "CNR International hit with enforcement notice over oil spill." 27 July 2018. <https://www.energyvoice.com/oilandgas/north-sea/177829/cnr-international-hit-with-enforcement-notice-over-oil-spill/>.

Thomas, L. Dale. "Selected systems engineering process deficiencies and their consequences." Acta Astronautica 61.1-6 (2007): 406-415.

Thompson, M. "Flying Shame." Time 2007: 34-44.

Topol, Eric. "Good Riddance to a Bad Drug." The New York Times 2 October 2004. 31 March 2015. <http://www.nytimes.com/2004/10/02/opinion/good-riddance-to-a-bad-drug.html>.

Traxler, Adrienne, et al. "Gender Fairness within the Force Concept Inventory." Physical Review Physics Education Research 14.1 (2018): 1-17.

Triantis, K., Rahmandad, H. and Vaneman, W. "Systems Engineering: Have we lost our competitive edge? A consideration of the dynamics of systems engienering projects." Albuquerque: International System Dynamics Conference, 2009.

TSBC. In-Flight Fire Leading to Collision with Water A98H0003. Gatineau: Transportation Safety Board of Canada, 1998.

—. Lac-Mégantic runaway train and derailment investigation summary. Gatineau: Transportation Safety Board of Canada, 2013.

Turner, Barry. Man-Made Disasters. London: Wykeham Science Press, 1978.

U.S. News. "Best Engineering Schools Ranked in 2018." 2018. 10 October 2018. <https://www.usnews.com/best-graduate-schools/top-engineering-schools/eng-rankings>.

United Nations Environment Programme and the Office for the Co-ordination of Humanitarian Affairs. Spill of Liquid and Suspended Waste at the Aural S.A. Retreatment Plant in Baia Mare. Geneva: United Nations Environment Programme and the Office for the Co-ordination of Humanitarian Affairs, 2000.

University of Illinois. "College of Engineering Undergraduate Majors and Minors." 8 October 2018. <https://engineering.illinois.edu/academics/undergraduate/majors-and-minors.html>.

University of Pennsylvania. "Majors & Minors." 8 October 2018. <https://admissions.upenn.edu/learning-at-penn/majors-minors>.

University of Southern California. "Engineering, Computer, Quantitative, and Technical Sciences." 8 October 2018. <https://undergrad.usc.edu/programs/major/list/engineering/>.

Vartabedian, Ralph and W. J. Hennigan. "Delays, technical glitches and huge cost overruns in the Air Force's F-22 fighter jet program highlight the Pentagon's broken procurement process." 16 June 2013. LA Times. <www.latimes.com/business/la-fi-advanced-fighter-woes-20130616-dto-htmlstory.html>.

Ward, Dan. "Opinion: How Budget Pressure Prompted the Success of Virginia-Class Submarine Program." 3 November 2014. USNI News. <news.usni.org/2014/11/03/opinion-budget-pressure-prompted-success-virginia-class-submarine-program>.

Warren, Tom. "Zune hardware was a mistake, admits former Microsoft exec Robbie Bach." 14 May 2012. The Verge. <www.theverge.com/2012/5/14/3019100/zune-hardware-mistake-robbie-bach>.

Weiss, Todd R. "United axes troubled baggage system at Denver airport." 10 June 2005. Computerworld. <https://www.computerworld.com/article/2556725/it-project-management/united-axes-troubled-baggage-system-at-denver-airport.html>.

Whoriskey, Peter. "Internal Ford documents about Explorer rollovers take a look at engineering." 8 May 2010. The Washington Post. <www.washingtonpost.com/wp-dyn/content/article/2010/05/08/AR2010050801571_pf.html>.

Wilcox, Joe. "iPod didn't kill Zune, Microsoft did." 2010. betanews. <betanews.com/2011/03/14/ipod-didn-t-kill-zune-microsoft-did/>.

Williams, Terry, et al. "Identifying and Acting on Early Warning Signs in Complex Projects." Project Management Journal 53.2 (2012): 37-53.

Winter, Donald C. "Getting Shipbuilding Right." U.S. Naval Institute Proceedings 133.6 (2007): 16-20.

WNA. "Chernobyl Accident 1986." October 2014. World Nuclear Association. <http://www.world-nuclear.org/info/Safety-and-Security/Safety-of-Plants/Chernobyl-Accident/>.

Wolverton, Troy and Dean Takahashi. "Microsoft's costly Xbox problem." 6 July 2007. Mercury News. <http://www.mercurynews.com/search/ci_6310888#>.

Woo, D M and K J Vicente. "Sociotechnical Systems, Risk Management, and Public Health: Comparing the North Battleford and Walkerton Outbreaks." Reliability Engineering & System Safety 80.3 (2003): 253-269.

Wright, Joshua. The Most In-Demand (And Aging) Engineering Jobs. 2014. 5 March 2018. <https://www.forbes.com/sites/emsi/2014/09/12/the-most-in-demand-and-oldest-engineering-jobs/#5f6bf5901e37>.

Yam, Marcus. "Microsoft Insider: Xbox 360 Team Knowingly Launched Faulty Hardware." 24 January 2008. DailyTech. <www.dailytech.com/Microsoft+Insider+Xbox+360+Team+Knowingly+Launched+Faulty+Hardware/article10434.htm>.