Applying Systems Thinking to Engineering and Design

Jamie P. Monat * and **Thomas F. Gannon**

Systems Engineering Program, Worcester Polytechnic Institute, Worcester, MA 01609, USA; tgannon@wpi.edu
* Correspondence: jmonat@wpi.edu

Abstract: The application of Systems Thinking principles to Systems Engineering is synergistic, resulting in superior systems, products, and designs. However, there is little practical information available in the literature that describes how this can be done. In this paper, we analyze 12 major Systems Engineering failures involving bridges, aircraft, submarines, water supplies, automobiles, skyscrapers, and corporations and recommend Systems Thinking principles, tools, and procedures that should be applied during the first few steps of the System Engineering design process to avoid such catastrophic Systems Engineering failures in the future.

Keywords: systems thinking; systems engineering; design

1. Introduction

1.1. Systems Thinking and Systems Engineering

Systems Thinking and Systems Engineering are not the same. Systems Thinking has been characterized as a perspective, a language, and a set of tools [1]. It is a holistic perspective that acknowledges that the relationships among system components and between the components and the environment are as important (in terms of system behavior) as the components themselves. It is a language of feedback loops, emergent properties, complexity, hierarchies, self-organization, dynamics, and unintended consequences. Systems Thinking tools include the Iceberg model which posits that, in systems, repeated events and patterns (which are observable) are caused by structure (stocks, flows, and feedback loops), which are, in turn, caused by underlying forces such as mental models, gravity, and electromagnetism. Additional Systems Thinking tools include causal loop diagrams, behavior-over-time plots, stock-and-flow diagrams, systemic root cause analysis, dynamic modeling tools, and archetypes. A more comprehensive explanation of Systems Thinking is provided by Monat and Gannon [1].

Systems Engineering is an interdisciplinary approach and means to enable the development of successful systems. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, and then proceeding with design, synthesis, validation, deployment, maintenance, evolution and eventual disposal of a system. Systems Engineering integrates a wide range of engineering disciplines into a team effort, which uses a structured development process that proceeds from an initial concept to production and operation of a system. It takes into account both the business and technical needs of all customers with the goal of providing a quality product that meets the needs of all users. A more comprehensive description of Systems Engineering and the structured processes used by Systems Engineers to develop systems is provided by Kossiakoff et al. [2].

Table 1 shows the Systems Thinking concepts that are applicable to Systems Engineering and Design.

Table 1. Systems Thinking concepts applicable to Systems Engineering and Design.

Systems Thinking Principle	Applicable to Systems Engineering and Design?
Holistic Perspective/Proper Definition of System Boundaries	Yes
Focus on Relationships	Yes
Sensitivity to Feedback	Yes
Awareness that Events and Patterns are caused by Underlying Structures and Forces	To a Degree
Dynamic Modeling	Sometimes
Systemic Root Cause Analysis	Yes, when troubleshooting
Sensitivity to Emergent Properties	Yes
Sensitivity to Unintended Consequences	Yes

Systems Thinking and Systems Engineering are related and synergistic, and the application of Systems Thinking Principles to Systems Engineering can result in superior systems. However, there seems to be a dearth of practical information describing how this can be done.

Lawson [3] discusses the natural coupling of thinking and engineering systemic structures, and gets into the specifics of applying Systems Thinking principles and tools (such as a System Coupling Diagram) to Systems Engineering. He couples thinking with acting via OODA (Observe-Orient-Decide-Act) and PDCA (Plan-Do-Check-Act) loops [4,5]. The PDCA loop seems especially relevant to systems engineering. Lawson provides a good example of the application of Systems Thinking principles to the development of automatic train control in Sweden in the 1970s. Kasser and Mackley [6] do a very nice job describing various Systems Thinking Perspectives (Operational, Functional, Big picture, Structural, Generic, Continuum, Temporal, Quantitative, Scientific) and applying them (by way of example) to the Royal Air Force (RAF) Battle of Britain Air Defence System (RAFBADS) by asking "Who, what, why, where, when, and how?" for each perspective. The Systems Thinking perspectives represent a good tool; however, more tools and specifics would be useful in inculcating Systems Thinking into Engineering and Design. Godfrey [7] suggests that Systems Thinking is essential for Systems Engineers and that Systems Thinking provides a key underpinning for Systems Engineering. Godfrey's work supports Kasser's "Who, what, why, where, when, and how?" approach but focuses on explaining Systems Thinking as opposed to its application to and integration with Systems Engineering. Pyster et al. [8] obliquely relate Systems Thinking to Systems Engineering via the application of patterns and archetypes. The INCOSE Systems Engineering Handbook [9] describes Systems Thinking and lists several Systems Thinking principles that Systems Engineers are likely to encounter. It does not, however, prescribe a methodology or integrate Systems Thinking tools with Systems Engineering. Burge [10] extols the virtues of Systems Thinking in engineering design and points out that it may be used in several different ways: to gain understanding of a complex situation, to gain sufficient understanding to make predictions of future system behavior, to solve a problem, or to create a new or modified system; however, he does not explain *how* this is to be done. Frank [11] explores the cognitive competencies (several of which involve Systems Thinking) of successful Systems Engineers, but he does not discuss how Systems Thinking principles or tools should be applied to Systems Engineering.

There are also instructive examples of Systems Engineering failures due to inattention to Systems Thinking principles.

1.2. Systems Engineering Failures Due to Lack of Systems Thinking

There have been many Systems Engineering and design failures due to a lack of applying Systems Thinking. In this section, we describe a few of the more infamous cases.

Galloping Gertie. The infamous Tacoma Narrows Bridge linked Tacoma, Washington to the Kitsap Peninsula from 1938–1940. The bridge spanned the Tacoma Narrows, which was known for high winds. Those high winds caused both an up-and-down roller-coaster like oscillation and aeroelastic *torsional flutter*: an oscillatory twisting of the bridge deck [12]. This latter phenomenon comprised a destructive reinforcing feedback mechanism when winds blew horizontally across the bridge deck (similar to the sound-inducing vibration created by blowing across a taught blade of grass). When the windward edge of the deck flexed slightly up or down (in this case due to a support cable failure), more of the deck's horizontally-projected surface area was exposed to the wind, yielding higher wind forces, which twisted the deck even more. The twist increased until the deck's torsional restoring force returned it to horizontal *and beyond*, and the process repeated, yielding a sinusoidal torsional oscillation (several good videos of this phenomenon and the subsequent collapse may be found online.) Vortex shedding downwind of the deck exacerbated the oscillations. The torsional oscillations were usually adequately damped by the stiffness of the bridge deck, but with the unfortunate combination of high wind speed and a snapped support cable (occurring on 7 November 1940), the damping declined to near zero. At this point, the torsional oscillations grew ever larger, until the bridge collapsed (Figure 1).

Figure 1. The Tacoma Narrows Bridge collapse.

Specific System Thinking Oversight: Failure to bound the system properly and to adequately evaluate system component interactions with the environment under all conditions, especially with respect to feedback. The original bridge design had called for 25-foot high trusses under the deck, which would have stiffened the deck sufficiently to prevent the collapse. However, to save money, that design was supplanted with a new design that replaced the trusses with 8-foot high steel plates, saving $3 million, but significantly reducing torsional resistance. The bridge designers might be excused had there been no prior experience with wind-induced bridge failure and torsional flutter. However, these phenomena had both been observed in the late 19th century [13]; in this case, they were simply forgotten or ignored [14].

Millennium Bridge, London. The Millennium Bridge (Figure 2) is a pedestrian bridge crossing the Thames River in London. It was built from 1998–2000 and opened to pedestrian traffic in 2000. The bridge design (the result of a competition) was novel, with a shallow profile and the supporting cables *below* the deck [15]. Shortly after opening, pedestrians noticed a peculiar lateral swaying motion that caused them to naturally adjust their stride in synchronization with the sway [16]. This synchronization amplified the sway as the bridge was driven at its natural *lateral* resonance frequency in a reinforcing feedback loop (the sway caused pedestrians to march in step with the sway; the marching in step amplified the sway.) Vertical resonance caused by wind or by soldiers marching in step was well-known to bridge designers, but lateral resonance had not been anticipated in this new, modern design [17]. Eighty-nine vertical and horizontal dampers were installed on the bridge to correct this problem, at a cost of ~$7 million.

Figure 2. London's Millennium footbridge.

***Specific System Thinking Oversight*:** Failure to adequately evaluate system component interactions under all use conditions, especially with respect to feedback. Despite thorough engineering analysis prior to construction (including modeling and wind tunnel testing to assess both lateral and torsional impacts of *vertical* pedestrian excitation), the analyses did not consider synchronous *lateral* excitation. There was reference to this phenomenon in the literature [18,19], but building codes did not mention it. Subsequent to the bridge's construction, BD 37/01 (the British Standard on bridge live loading) was revised to include a section on synchronous lateral excitation.

The Lockheed L-188 Electra Turboprop Airplane. The L-188 Electra was a 4-engine turboprop airplane developed by Lockheed in the late 1950s. The plane suffered two fatal crashes in 1959 (Braniff 542) and 1960 (Northwest 710), killing 97 passengers and crew. In each of these crashes, excessive wing flutter and vibration led to the wings shearing off the fuselage. Analysis and wind tunnel tests revealed a fatal reinforcing mechanical feedback loop involving the plane's engine mounts. It was determined that the engine mounts permitted small oscillations of the engine on the wing. These engine oscillations caused the wing to flutter, which, in turn, caused the engine to oscillate more. The increased engine oscillation increased the wing flutter. This reinforcing feedback loop caused the wings to vibrate at their natural resonant frequency, eventually breaking away from the fuselage. In this situation, mechanical feedback was not considered in either the design stage or the engineering test stage. It is interesting to note that, after the crashes, the engine oscillations were subsequently reproduced in wind-tunnel testing of scale models.

***Specific System Thinking Oversight*:** Failure to properly evaluate system component interactions under all use conditions, especially with respect to feedback.

The Water of Ayolé. The Water of Ayolé demonstrates that few engineering/technical issues are exclusively technical; most involve support infrastructures, people, the environment, economics, and other factors. Ayolé is a small rural village in the West African country of Togo. In the 1970s–80s, the water source for the village was the Amou River, which happened to be infested with the guinea worm *Dracunculus medinensis,* a parasite that infects a human host and causes excruciating pain. To address this issue, government and international aid organizations dug and installed wells in the village, which worked well for several years. However, as the wells broke down (due to normal wear and tear), no spare parts were available, no technical expertise was available to fix or maintain the pumps, and no money was available to pay for repairs. After three years, the people of Ayolé were back to using the contaminated water from the river. The government engineers had interpreted this as a purely technical/engineering problem, when in fact it was much broader. To their credit, the local Togolese extension agents applied Systems Thinking to address the larger systemic issues. They trained some of the villagers in well maintenance and repair; they established a repair parts supply chain via the local Togo hardware store; and the women of the village structured an agricultural product production and sales system to generate money to pay for the parts. What was thought to be a simple engineering problem turned out to be an engineering/socio-economic/logistics/psychological problem.

Specific System Thinking Oversight: The assumption that real-world engineering problems can be solved by purely technical means. Many involve sociological, psychological, economic, legal, maintenance, support, and "soft" human issues.

Stow Center School Aquarium. A situation similar to that of Water of Ayolé occurred on a much smaller scale in 2003 at the Center School in Stow, MA, USA. A local university had donated a salt-water aquarium to the school, complete with filtration equipment, temperature regulation, and fish. The students loved the aquarium and were fascinated by its horseshoe crabs and other exotic-looking sea creatures, enjoying it for several years. However, then, components began to wear out and needed replacement, the system needed cleaning and maintenance, and the creatures required regular feeding and assessment. No one had been assigned the responsibility for system maintenance, and teachers (who often felt underpaid) did not feel that maintenance was their responsibility. Eventually, the beautiful, free aquarium was disassembled and scrapped.

Specific System Thinking Oversight: Like the Water of Ayolé, there were ancillary support and maintenance issues associated with the aquarium; issues rarely have purely technical solutions. The well-intentioned donors did not realize that this "gift" imposed a burden on the recipients.

The Russian K-141 Kursk Submarine Disaster. On 12 August 2000, the Russian Kursk nuclear submarine exploded and sank off the coast of Russia in the Barents Sea; all 118 crewmen were lost. The explosion was traced to a leak of hydrogen peroxide (H_2O_2) from one of the ship's torpedoes; the peroxide reacted explosively with copper or brass that was present in the torpedo tube.

Both hydrogen peroxide alone and the combination of kerosene and hydrogen peroxide had been used as propellants for rockets and torpedoes since the 1930s [20]. It was well-known that hydrogen peroxide reacts violently in the presence of a silver, copper, or brass catalyst, and previous near-disasters with H_2O_2-powered torpedoes had been well-documented. Because of this risk, hydrogen peroxide/kerosene propellants have been banned by the British and other navies and replaced by newer, safer combinations such as Otto Fuel and Hydroxyl Ammonium Perchlorate [21,22]. When the peroxide leaked from the torpedo and contacted the catalyst on the Kursk, it was converted to oxygen and water vapor, increasing in volume by a factor of 5000. This sudden pressure surge caused a subsequent explosion (equivalent to 500 pounds of TNT) in a nearby kerosene tank and blew a hole in the submarine's hull. It is thought that the sub then sank within minutes. A few minutes later (and while on the ocean floor), the heat from the kerosene explosion resulted in the detonation of several nearby torpedoes [23]. The explosions blew an immense hole in the sub, and most of the sub's compartments flooded. Subsequent confusion and lack of transparency by the Russians prevented a rescue of the trapped sailors.

Specific System Thinking Oversight: Failure to identify both system component planned inter-relationships and unplanned inter-relationships. The knowledge of the catalytic reaction of hydrogen peroxide with metals (with potentially devastating consequences) should have prevented the co-location of any hydrogen peroxide containers with metal catalysts.

The Vdara Hotel, Las Vegas. Las Vegas's Vdara hotel (Figure 3) was designed by Rafael Viñoly and built in 2008. The hotel's curved façade focuses the sun's rays like a parabolic reflector, heating the pool area at its base to over 135 °F—an unintended consequence [24,25]. Employees and visitors refer to the effect as the "Death Ray." Non-reflective film has been applied to the hotel's highly-reflective windows and large umbrellas have been placed around the pool area, but the deck still gets hot. The area gets so hot, in fact, that guests have reported burning skin and singed hair within minutes of lying near the focus [26].

Specific System Thinking Oversight: Failure to include relevant environmental components such as the sun and its interaction with other system components.

Figure 3. The Vdara Hotel.

20 Fenchurch Street, London. In London's financial district, 20 Fenchurch Street (Figure 4) rises 34 stories from the street. Also designed by Rafael Viñoly and completed in 2014, the parabolic shape of the highly reflective upper stories focuses sunlight onto a small area at street level for several hours each day, resulting in storefront temperatures exceeding 200 °F. An automobile was partially melted [27,28] and a reporter fried an egg on the sidewalk [29]. Local businesses were negatively affected by the intense heat. The thermal behavior caused locals to nickname the building the "Walkie-Scorchie" [30] and the "Fryscraper" [31]. Louvers, shades, and non-reflective glass have been considered as remediation measures.

Figure 4. 20 Fenchurch Street, London.

Specific System Thinking Oversight: One would think that Viñoly would have learned his system definition lesson after the Vdara debacle several years earlier. Such was not the case. Viñoly repeated the error of not including all relevant environmental components and their interrelationships in his design—in this case, the interaction of the sun with the building.

Toyota Gas Pedal/Floor Mat Entrapment. In 2007–2010, several accidents and fiery deaths were attributed to "sudden acceleration" in Toyota vehicles. Camrys, Avalons, Highlanders, Matrixes, Priuses, Venzas, Tacomas, and Tundras, and Lexus ES350s, IS250s, and IS350s were affected [32]. The problem was at first denied, and then traced to unintended interactions between the vehicles' floor mats and accelerator pedals [33]. It seems that in certain circumstances, the gas pedal would "bond" to

the floor mat resulting in unintended acceleration, inability to slow or stop the vehicle, and consequent accidents. The company recalled 4.2 million vehicles for floor mat replacements and potential gas pedal redesign.

Specific System Thinking Oversight: This is an interesting case (similar to the Kursk) in which the system components were properly identified, but the unintended interactions among components were not.

Biodegradable German Car Wiring Insulation. The Green Party in Germany passed a law in the early 1990s that required a certain percentage of the parts in an automobile to be bio-degradable, and the EU followed suit in the mid-1990s. Mercedes-Benz decided to rely on bio-degradable wiring insulation to meet those requirements. Unfortunately, a bio-degradable wiring system that is exposed to the environment will eventually decompose into a mass of short-circuiting copper wires [34].

Specific System Thinking Oversight: This situation is similar to the Galloping Gertie fiasco in which environmental factors (weather) were not adequately considered.

The Bhopal Disaster. A Union Carbide pesticide plant in Bhopal, India resulted in 2259 immediate deaths and some 11,000 delayed deaths following an accident in 1984. A highly toxic material called Methyl isocyanate, used in the making of pesticides, became contaminated with water, which caused an exothermic reaction that increased the temperature inside a tank well beyond its capacity. An automated emergency release system vented the extra pressure and a large volume of gasses escaped and spread to the surrounding town, killing over 13,000. If one were to define the "system" as just the pesticide plant, and consider that system in isolation, then the safety sub-system worked pretty well, relieving the pressure and preventing an explosion. However, when including the surrounding town and people in the definition of the "suprasystem" [35], it was an utter disaster. (A "suprasystem" is a larger system that integrates several smaller systems. In our usage here, it means the system proper plus the environment, system users, system controllers and maintainers, communications to and from the system, and power to the system.) Additionally, if the vented gasses had been lighter than air, they might have dispersed without much harm. Unfortunately, those gasses were heavier than air, and seeped into the nearby city of Bhopal at ground level. That leak caused over 550,000 injuries.

Specific System Thinking Oversight: This situation is similar to the Galloping Gertie fiasco in which the interactions of the system components with each other and with the environment (people in the surrounding town, weight of gases) were not considered adequately.

The Microsoft Zune. In response to Apple's fabulously successful iPod (released in 2001), Microsoft released its own portable music player, the Zune, in 2006. The Zune did not have the iPod's aesthetic appeal or "cool" factor [36]. Perhaps more significantly, Microsoft did not appreciate that the Zune and all personal media players are part of a *User Experience System,* and that, to be successful, all components of that system must be addressed. Steve Jobs and Apple were fabulous at this. Don Norman says, "It is *not* about the iPod; it is about the **system**. Apple was the first company to license music for downloading. It provides a simple, easy to understand pricing scheme. It has a first-class website that is not only easy to use but fun as well. The purchase, downloading the song to the computer and thence to the iPod are all handled well and effortlessly. In addition, the iPod is indeed well designed, well thought out, a pleasure to look at, to touch and hold, and to use. There are other excellent music players. No one seems to understand the systems thinking that has made Apple so successful" [37]. Although Microsoft eventually attempted to develop a computer interface, a music subscription service ("Zune Music Pass"), and download capabilities ("MSN Music" and "Zune Marketplace"), those efforts were too little, too late, and the *User Experience System* was clearly not Microsoft's focus. The Zune failed commercially and was discontinued in 2011. It should be noted that other portable music device manufacturers (Sony, Diamond, Tascam) also missed the systems thinking aspects of product design: engineering the product is not the same as engineering the *User Experience System*.

Specific Systems Thinking Oversight: Failure to understand that many products are not really stand-alone devices, but instead are merely one component of a *User Experience System*.

These failures demonstrate the need to apply Systems Thinking principles (especially appropriate "system" definition, system boundaries, identification of all relevant system *and environmental* components and relationships, and consideration of feedback mechanisms) to Systems Engineering and design. They are summarized in Table 2.

Table 2. Summary of 12 Engineering/Design problems deriving from poor Systems Thinking.

Problem	Systems Thinking Issue
Galloping Gertie	Failure to adequately address planned and unplanned interactions between system components and the environment.
Millenium Bridge, London	Failure to adequately address planned and unplanned interactions among system components themselves and between system components and the environment.
Lockheed L-188 Electra Turboprop Airplane	Failure to adequately address planned and unplanned interactions among system components themselves and between system components and the environment.
Water of Ayolé	Failure to bound the system properly; specifically, to understand that most complex problems cannot be solved by purely technological means; they often involve organizational, political, economic, environmental, ethical, and sociological components.
Stow Center School Aquarium	Failure to bound the system properly; specifically, to understand that most complex problems cannot be solved by purely technological means; they often involve organizational, political, economic, environmental, ethical, and sociological components.
Russian K-141 Kursk Submarine	Failure to adequately address planned and unplanned interactions among system components themselves and between system components and the environment.
Vdara Hotel	Failure to identify relevant environmental factors.
20 Fenchurch Street, London	Failure to identify relevant environmental factors.
Toyota Gas Pedal	Failure to adequately address planned and unplanned interactions among system components themselves and between system components and the environment.
Biodegradable German Car Wiring Insulation	Failure to identify relevant environmental factors.
Bhopal, India	Failure to identify relevant environmental factors; specifically, interactions between the system and neighboring people and between the system and environmental gases.
Microsoft Zune	Failure to recognize that many products are actually components of a *User Experience System*

It is interesting that the Systems Thinking failures explaining these 12 problems fall into just four categories:

a. Failure to identify relevant environmental factors such as wind, insolation, rain, and temperature;
b. Failure to understand that most complex problems cannot be solved by purely technological means; they often involve organizational, political, economic, environmental, ethical, and sociological components.
c. Failure to adequately address both planned and unplanned interactions among the system components themselves and between system components and the environment.
d. Failure to recognize that many products are actually components of a *User Experience System.*

2. How to Engineer and Design Systems to Avoid Similar Issues

2.1. Identification of Relevant Environmental Factors such as Wind, Insolation, Rain, Temperature

Failure to identify a system's relevant environmental factors can result in disaster. In the Biodegradable German Car Wiring Insulation debacle, failure to consider normal weather impacts on the insulation caused catastrophic failures. In the Vdara hotel situation (and the 20 Fenchurch Street example), failure to consider the interaction of the building with the sun yielded poor results.

It is sometimes difficult to decide what is part of a system versus what is part of the system's environment. Kossiakoff [2] suggests four criteria to decide which elements are system versus environmental components:

1. Developmental Control: the ability of the system designer to control the component;
2. Operational Control: the ability of the system operator to control the component;
3. Functional Allocation: the ability of the system designer/operator to assign functions to the component;
4. Unity of Purpose: the degree to which the component is dedicated to the system's successful performance.

Components for which these four criteria score low are defined as environmental (not system) components. Kossiakoff suggests depicting system boundaries using a *Context Diagram* (see Figure 5, adapted from Johns Hopkins [38]).
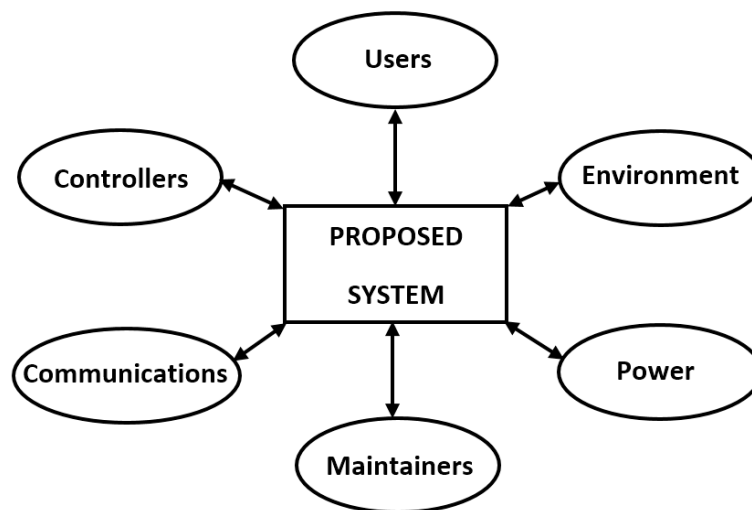
Figure 5. System context diagram [38].

The context diagram notes additional entities (environment, users, maintainers, etc.) that must be considered in system design. For present purposes, we will focus on the environment. To identify environmental factors that may impact system performance, more detail and specificity are required for both system and environmental components. An excellent tool for this is the *System Breakdown Structure* or SBS.

An SBS is a hierarchical pictogram showing the systems, environment's, and user's components (see Figure 6). (Subsequently, the interrelationships among these components will be described; but first the components themselves must be identified.) Additional sublevels may be added to whatever degree is necessary. For example, Sub-System 1 may comprise several sub-sub-systems. Often, non-Systems Thinkers focus only on the tangible physical components of a system; this can lead to problems as made evident by Table 2. Note that Figure 6 represents a good starting point for an SBS; however, it should be tailored for each specific system design. For example, designers of offshore oil rigs would need to include more environmental factors related to seas, the ocean floor, and underwater life. The top levels of the SBS correspond to the elements depicted in Figure 5; however, the SBS contains much more detail and a level of specificity that is actionable.
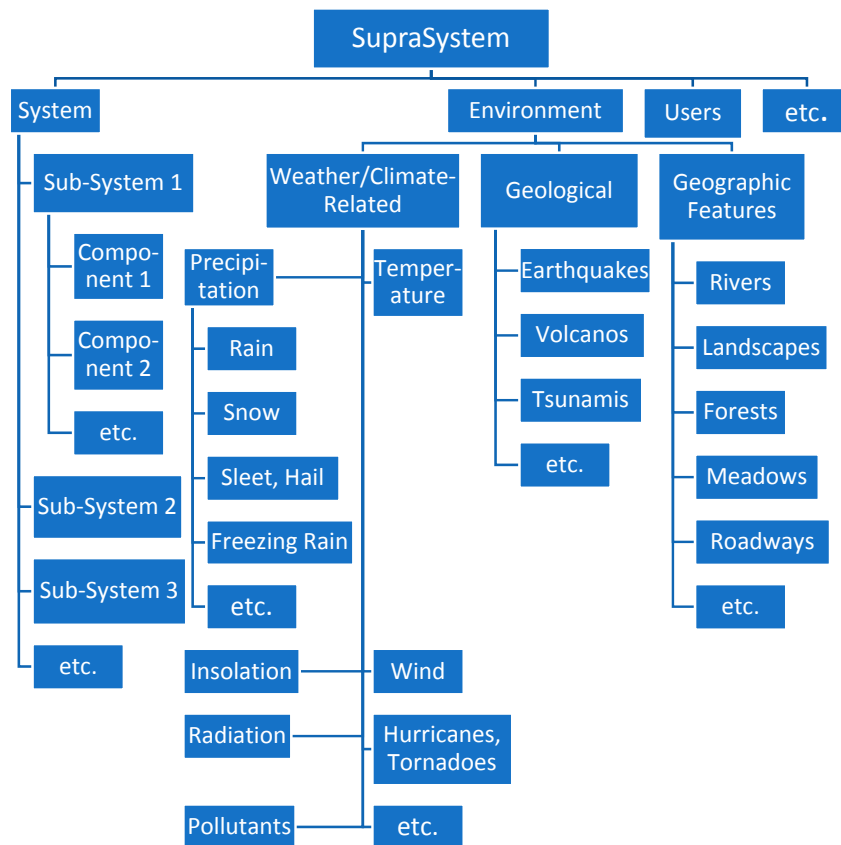
Figure 6. System breakdown structure (SBS).

2.2. Bounding the System Properly with Respect to Understanding That Most Complex Problems Cannot Be Solved by Purely Technological Means; They Often Involve Organizational, Political, Economic, Environmental, Ethical, and Sociological Components

In the Water of Ayolé example, a narrow perspective of the problem as only an engineering issue led to failure. This was repeated in the Stow Center School aquarium situation. It is a common oversight. Few technical issues do not have human, environmental, economic, sociological, or emotional issues associated with them, yet many engineers are uncomfortable dealing with these "softer" aspects of engineering.

In Ayolé, the government assumed that installing pumps would solve the water problem. However, three years after installation, the pumps were no longer functional. The following sub-systems needed to be created and installed to fully solve the problem:

i.   A water and pump operation training and education system;
ii.   A cultural sensitivity system;
iii. A pump maintenance and repair system;
iv.   A supply chain for repair parts;
v.   A money-generation and management system, including a farming sub-system;
vi.   A village/social organization to appropriately divide the labor and decision-making.

It is not hard to generalize from these examples when designing or engineering a new or modified system:

i.   Will the new system or systemic change require any cultural adjustment? If so, cultural sensitivity and training will be required.
ii.   Will the new system require training and education in its use, benefits, and maintenance?
iii. Will a cleaning, maintenance, and repair system need to be established?

iv.   Will a repair parts supply chain need to be established?

v.   Is there a means to pay for repair, maintenance, legal issues, decommissioning?

2.3. Adequately Addressing Both Planned and Unplanned Interactions among the System Components Themselves and between System Components and the Environment

Several of the examples listed (Kursk, Galloping Gertie, Lockheed Electra, Millennium Bridge, Toyota Gas Pedal, Biodegradable German Car Wiring, Bhopal) resulted from failure to identify potential interactions among system components, or between system components and the environment. Several tools are available to minimize the probability of overlooking these relationships:

2.3.1. System Interrelationship Matrix

One of the best tools for identifying planned and unplanned relationships among system components and among system and environment components is the System Interrelationship Matrix or SIM. One constructs the SIM by listing all system and environmental components on both axes of a two-dimensional matrix, as shown in Figure 7. Then, one places an X in every cell representing an interaction between the components. One may add detail by noting (in addition to the X) the type of interaction: for example, command-control, mechanical, chemical, emotional/psychological, organizational, frictional. One may make a single matrix for the entire suprasystem, or one may develop smaller, more tractable SIMs for sub-levels or components, as shown in Figure 8 for automobile components.

	Engine	Brakes	Trans-mission	Suspen-sion	Ignition System	Fuel System	Driver	Passen-gers
Engine	-		X		X	X		
Brakes		-		X			X	
Trans-misson	X		-	X			X	
Suspension		X	X	-				
Ignition System	X				-			
Fuel System	X					-		
Driver		X	X				-	X
Passengers							X	-

Figure 7. SIM for the highest level of an automobile.

Suspension Sub-System	Wheels	Tires	Springs	Shocks and Struts	Linkages	Bushings, Bearings, Joints	Steering System
Wheels	-	X	X	X	X	X	X
Tires	X	-					
Springs	X		-	X	X	X	X
Shocks and Struts	X		X	-	X	X	X
Linkages	X		X	X	-		
Bushings, Bearings, Joints	X		X	X		-	X
Steering System	X		X	X		X	-

Figure 8. SIM for the components of an automobile suspension.

System Interrelationship matrices have the great benefit of comprehensiveness; however, they can become unwieldy.

2.3.2. Stock-and-Flow Diagrams

Stock and Flow diagrams depict the interrelationships among system components, and between systems and their environments, from a control volume perspective. Monat and Gannon [39] state, "In systems, some quantities are stored while others flow. These may be real physical quantities such as dollars, volume of water, number of customers, or number of cabbages in a field. They may also be non-physical quantities such as love, anger, greed, or other emotions. Stores or accumulations of these items are called "stocks". Stocks increase or decrease as quantities flow into or out of them. Like causal loop diagrams, stock-and-flow diagrams are helpful in understanding systemic behavior." An example of a stock-and-flow diagram is shown in Figure 9.
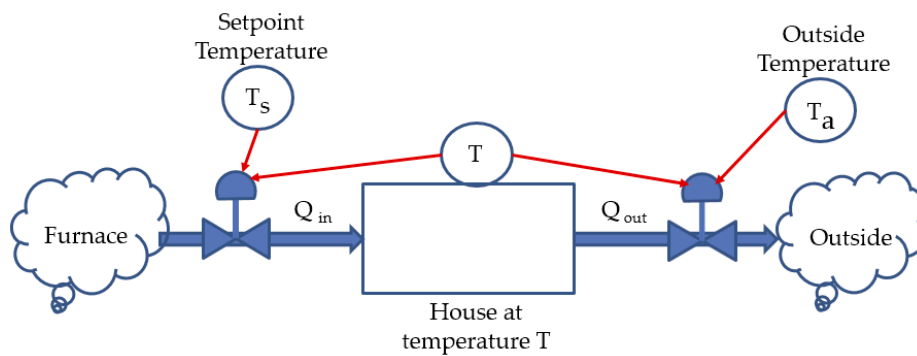
Figure 9. Stock-and-flow diagram for domestic heating system control.

A weakness of stock-and-flow diagrams is that it is hard to determine if one has been comprehensive in showing *all* stocks and flows.

2.3.3. Causal Loop Diagrams

Causal Loop Diagrams (CLDs) are another tool that may be used to show cause-and-effect relationships among system and environmental components. They are especially helpful in depicting feedback processes, which are present in most systems. Some feedback loops (such as the stabilizing feedback mechanism of a Proportional-Integral-Differential (PID) controller or the reinforcing feedback mechanism of compound interest) are obvious, whereas some (financial bailouts, trade tariffs) are more subtle.

An extremely simple version of a domestic heating system CLD is shown in Figure 10.
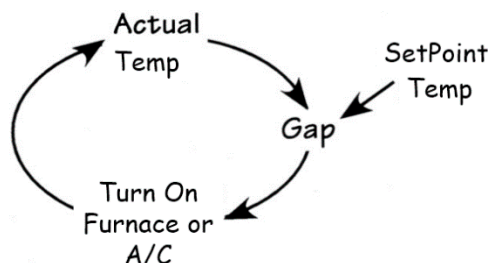

Figure 10. Domestic heating system causal loop diagram.

CLDs can become complicated as various cause-and-effect relationships are identified and depicted. A more complicated CLD (related to the Water of Ayolé example) is shown in Figure 11. This CLD highlights the interactions among the water system and the sociological impacts of the system.
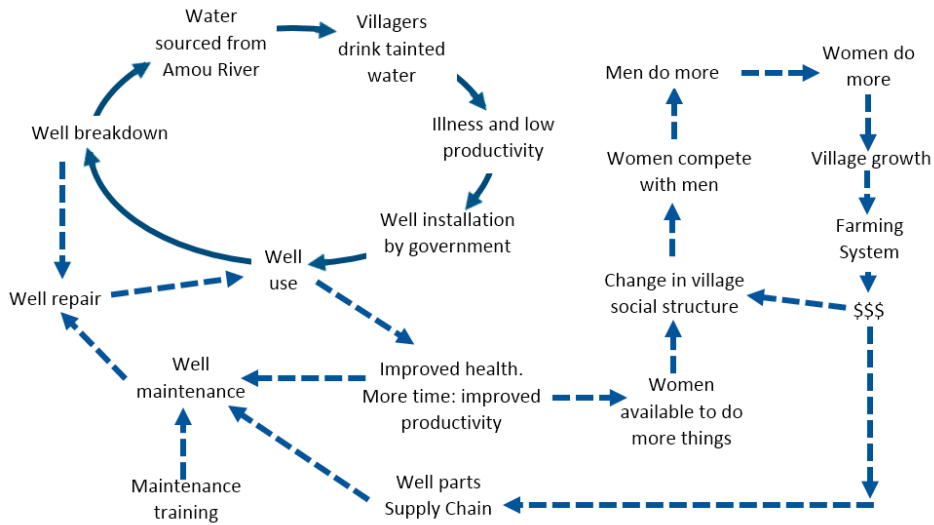
Figure 11. CLD for the Water of Ayolé Example. The solid arrows represent the initial effort; the dashed arrows represent the additional structure after the final effort.

As for stock-and-flow diagrams, CLDs are useful, but it is hard to determine if one has been comprehensive in capturing all interrelationships. CLDs are explained in greater detail in *The Systems Thinker* [40] and in Kim [41].

2.3.4. N^2 Diagrams

An N^2 chart or N^2 diagram is an N x N matrix designed to show interfaces based on system function. Various system functions are plotted on the matrix diagonal; inputs are shown vertically (up or down) and outputs are shown horizontally (left or right). A generic N^2 diagram is shown in Figure 12 while a specific one is shown in Figure 13.
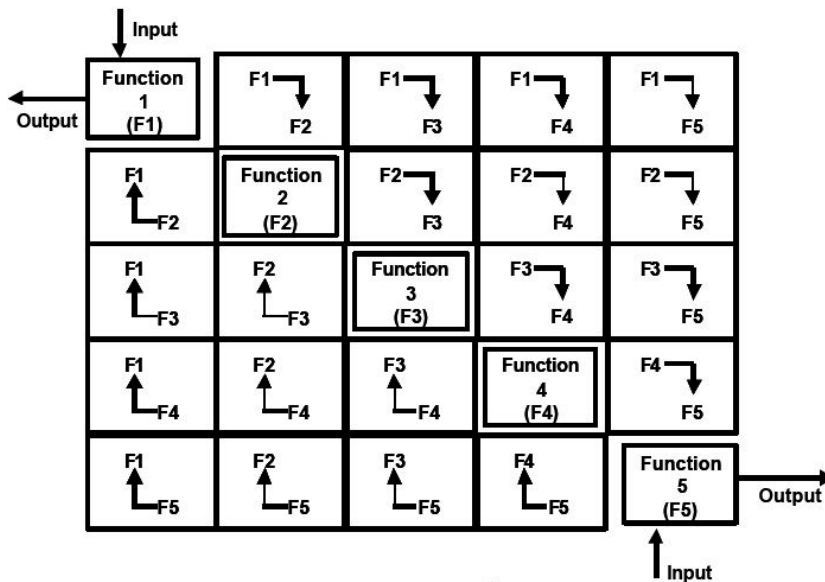

Figure 12. Generic N^2 diagram from FAA [42].

Feedback loops may be shown as closed circles and critical functions are identified as cells in which several circles intersect. N^2 diagrams are useful constructs; however, it is not clear that they are comprehensive and identify *all* interactions. For example, they show only interfaces between functions; some system and environment components are not functions and therefore those interrelationships may be missed.
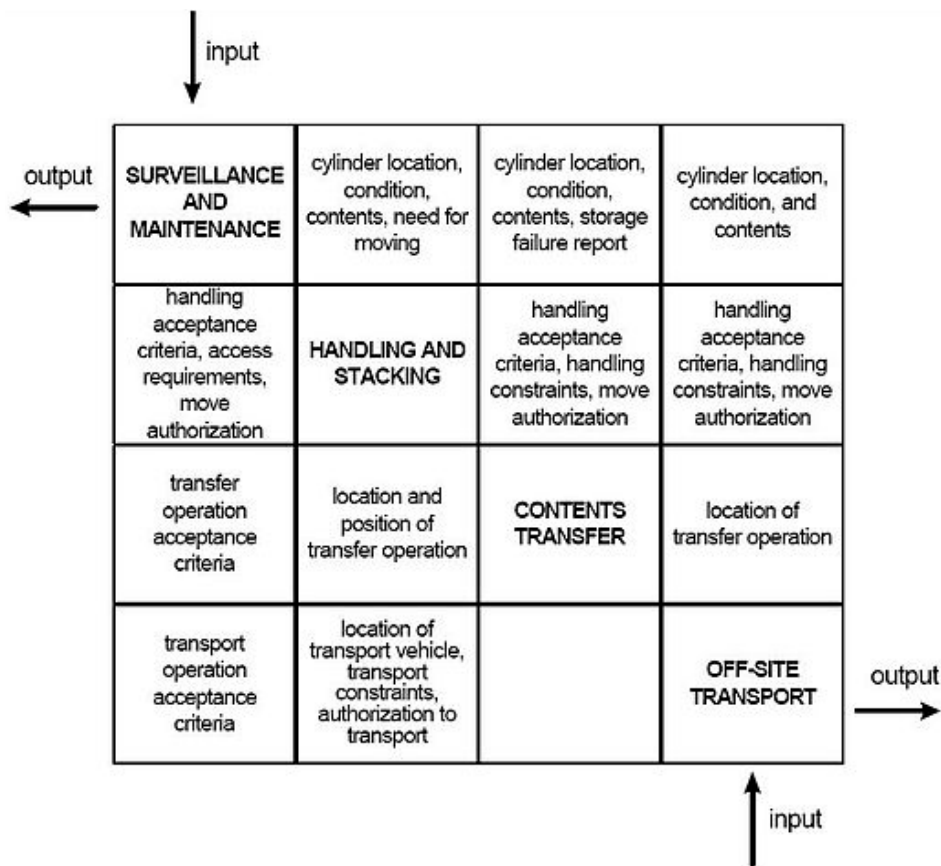
Figure 13. Specific N^2 diagram for "Functional Flow of Operations" for UF$_6$ cylinder management, from Bechtel Jacobs Company, (Oak Ridge, TN USA) [43].

2.3.5. SV-3 System-System Matrix

The SV-3 Systems-Systems Matrix is a Department of Defense (DoD) construct designed to summarize system resource interactions, manage interfaces, and compare interoperability characteristics of solution options [44]. The DoD website states, "The SV-3 is generally presented as a matrix, where the Systems resources are listed in the rows and columns of the matrix, and each cell indicates an interaction between resources if one exists. Many types of interaction information can be presented in the cells of a SV-3. The resource interactions can be represented using different symbols and/or color coding that depicts different interaction characteristics, for example:

a. Status (e.g., existing, planned, potential, de-activated);
b. Key interfaces;
c. Category (e.g., command and control, intelligence, personnel, logistics);
d. Classification-level (e.g., Restricted, Confidential, Secret, Top Secret);
e. Communication means (e.g., Rim Loop Interface, Scalable Loop Interface)."

An example of an SV-3 matrix, based upon the development of the Mobil SpeedPass, is shown in Figure 14.
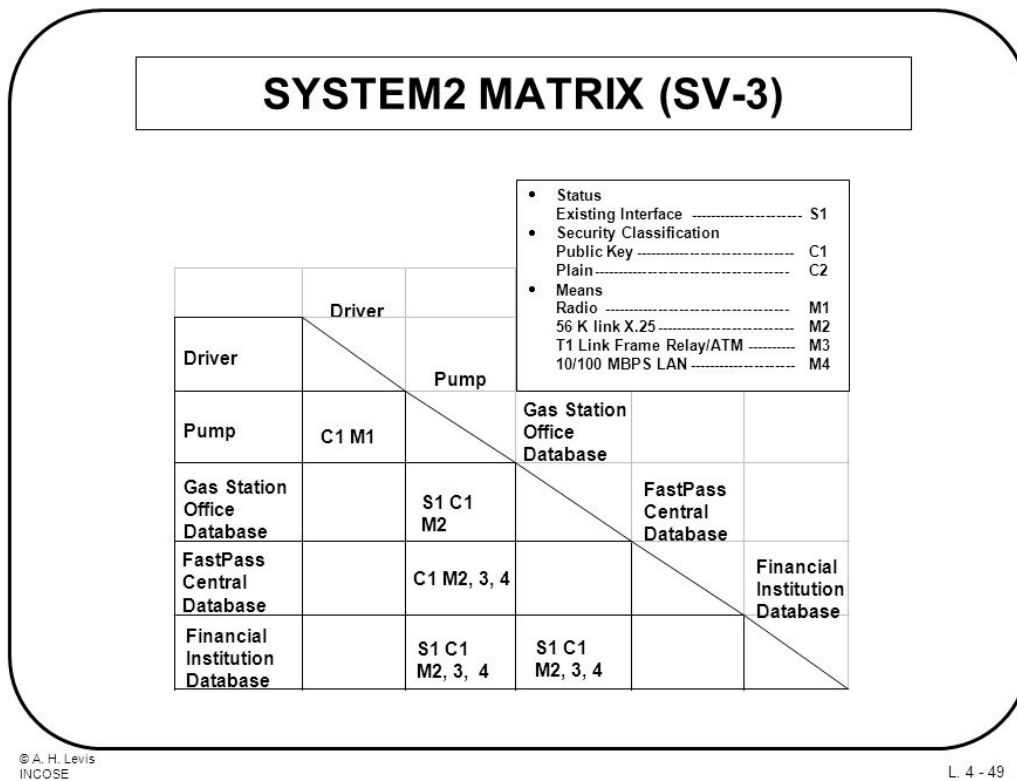
Figure 14. Mobil SpeedPass SV-3 matrix [45].

One weakness of the SV-3 is that it depicts interfaces only among systems and subsystems, not among components. Therefore, some key interrelationships may be missed.

2.4. Recognizing That Many Products Are Part of a Larger User Experience System

Steve Jobs, founder of Apple Computer was a great Systems Thinker. He recognized that most consumer products are not stand-alone products, but part of experiential *consumer use systems*. While competitors like Sony, Tascam, Microsoft, and Diamond structured their companies around their products (such as portable music players), Jobs structured Apple around the consumer's *experience*, of which the product (in this case the iPod) was just one component. The device itself is just one element of listening to music; other elements include the means of acquiring the music, the user's activities while listening, the environment while listening, and the prestige/coolness factor that may be associated with both the product and the listening experience [37] (see Figures 15 and 16).
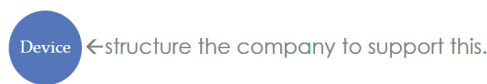

Figure 15. Sony, Tascam, Microsoft, Diamond Structure [39].

Figure 16. Apple structure [39].

The iPod (and in fact most Apple products) was developed using similar innovative Systems Thinking focused on the user's *experience* as opposed to the product. It is natural to ask if other products should be considered through this innovative Systems Thinking "experiencing" versus "owning" lens:

- The automobile as a stand-alone product versus the *car buying and owning experience*. Car ownership involves car purchase, registration, annual inspections, maintenance, and disposal or trade-in as well as insurance. Cars wear out and new technology renders older models obsolete. There is no good reason that car dealers could not provide all these services for a fixed monthly fee. Some dealerships have already started down this path with service areas that provide free meals, entertainment, and drop-off services. Several manufacturers (Volvo, Cadillac, BMW) have adopted new "subscription services" in which a fixed monthly fee is paid by the user to cover lease, insurance, maintenance, and other expenses [46].
- Coffee. Is it the coffee itself, or is it the *coffee-drinking experience*? Starbucks (and others) attracts clients to not only buy and drink coffee, but to enjoy the coffeehouse experience, with free Wi-Fi, comfortable seating, and even fireplaces in some establishments.
- Clothes versus the clothes *buying and owning experience*. Clothes use involves clothing selection, travel to a store, fitting, matching, laundering and pressing, repair, and disposal. While many do not mind (or even enjoy) these activities, some do not. Enterprising Systems Thinking businesses could assume all these functions for a fixed monthly fee, thus providing a clothes use *experience* in which the clothes themselves are merely adjuncts.
- Flat Panel TV versus the *home entertainment system experience*. To many people, the home theater component selection, purchase, matching, interconnection, and set up is a harrowing experience. A Systems Thinking approach by subscription TV service providers (Comcast, Verizon, DirecTV, DishTV, etc.) would dictate that these onerous functions be included in the monthly subscription. This would ensure that users would always have the latest equipment set up and functioning optimally to receive the provider's streaming content. It would be similar to a razor-blade or inkjet printer business model in which the asset (in this case the TV and associated hardware) is provided free or near-free to encourage the user to consume the razor blades or ink (in this case, the streaming services).
- Home Ownership. Certainly many houses are rented today; however, house *buyers* must assume the responsibility for lawn and yard maintenance, utilities, snow removal, pool maintenance, insurance, and all the other onerous responsibilities that come with home ownership. A Systems Thinking approach to the home owning experience would bundle these items in with monthly mortgage payments such that the owner pays one monthly fee for *all* home owning tasks and services, which are then provided by the mortgage company or their representatives.

There are other examples for which close inspection (from a Systems Thinking perspective) reveals that the *product* is really just a part of a *User Experience System*.

3. Procedure

The Systems Engineering procedure is described well by Blanchard and Fabrycki [47] and by Kossiakoff et al. [2]. The principle steps are:

1. Regional Architecture/Scoping the Problem;
2. Feasibility Study/Concept Exploration;
3. Concept of Operations;
4. System Requirements;
5. High-Level Design;
6. Detailed Design;
7. Software/Hardware Development/Field Installation;
8. Unit Device Testing;

9.　Subsystem Validation;

10.　System Verification and Deployment;

11.　System Validation;

12.　Operations and Maintenance;

13.　Changes and Upgrades;

14.　Retirement/Replacement.

Many of the engineering and design failures described earlier in this paper were the result of failures to bound the system properly, specifically interactions with environmental factors, and failures to adequately address planned and unplanned interactions among system components themselves and between system components and the environment. Those failures were the result of inadequate scoping of the problem (Step 1 above), which is directly related to the Systems Thinking concepts of Holistic Perspective/Proper Definition of System Boundaries and a Focus on Relationships. If the boundaries and interactions of a system with its environment are not defined properly at the onset of the Systems Engineering process, each of the subsequent steps in that process will be focused on incomplete or inaccurate requirements. The high-level and detailed design of the system will not take into account critical interactions of the system with its environment. Moreover, the Verification and Validation of the system will be incomplete and not consider those interactions of the system with its environment, which could result in catastrophic failure of the system.

To avoid similar failures from arising in the future, a holistic perspective must be taken at the beginning of the Systems Engineering lifecycle, and revisited often throughout that lifecycle. A wide range of relationships among many of the system components themselves and the environment in which the system is intended to operate must be considered. That range of relationships should also include organizational, political, economic, environmental, ethical, and sociological factors. The tools described above are instrumental in achieving this.

Finally, many engineered products (automobiles, appliances, tools, entertainment devices, clothing, prepared foods, engineered homes, etc.) are merely components of a *User Experience System*. Systems Thinking requires that engineers pay attention to all aspects of this system when designing products.

4. Conclusions and Recommendations

Engineers can sometimes be excused for problems that have never before been observed. However, we should learn from our mistakes. The problems associated with several of the examples listed in this paper had been previously observed and documented, but that information was either not researched, ignored, or (more likely) not available in an organized, actionable way. Systems Thinking helps organize some historical engineering and design errors into a set of principles that can be beneficial and actionable. Taking a holistic view and focusing on relationships (as opposed to system components) can minimize the chances of engineering and design gaffs. In this paper, we analyzed 12 Systems Engineering failures and found that they could be categorized into four types of Systems Thinking errors: failure to identify relevant environmental factors, failure to understand that most complex problems cannot be solved by purely technological means, failure to adequately address both planned and unplanned interactions among the system components themselves and between system components and the environment, and failure to recognize that many products are actually components of a *User Experience System*. Several tools are available to address these issues:

1.　Early in the design process, system engineers must ensure that they have captured all relevant components of the system, as well as the suprasystem in which the system of interest resides. Environmental factors such as wind, insolation, temperature, and the potential for dramatic incidents such as earthquakes, tsunamis, and hurricanes are notorious for being overlooked. The System Breakdown Structure (SBS) is an excellent tool for this.

2. Once the relevant components of the system of interest and its suprasystem have been properly identified, all relevant relationships must be identified. The System Interrelationship Matrix (SIM) is an excellent tool for this. Stock-and-Flow diagrams and Causal Loop diagrams may also be helpful.

3. Having identified all relevant system components and interrelationships, systems engineers must then realize that very few complex problems have purely technological solutions. Engineers also must consider the sociological, psychological, ethical, political, cultural, and economic factors that may impact the success of a complex system.

4. Finally, in designing complex systems, engineers must understand that "products" are very often not stand-alone devices or systems, but instead part of a *user experience system* that comprises the user, the environment, aesthetics, psychological factors, system acquisition, maintenance, upgrades, and disposal. Failure to address these ancillary factors may cause the system to fail, either technically or commercially.

We recommend that these tools be applied during the first few steps of the standard System Engineering design process to avoid catastrophic Systems Engineering failures in the future.

References

1. Monat, J.P.; Gannon, T.F. What is Systems Thinking? A Review of Selected Literature Plus Recommendations. *Am. J. Syst. Sci.* **2015**, *4*, 2.

2. Kossiakoff, A.; Sweet, W.; Seymour, S.; Biemer, S. *Systems Engineering: Principles and Practice*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2011.

3. Lawson, H. *A Journey Through the Systems Landscape*; College Publications: London, UK, 2010; p. 6.

4. Boyd, J.R. An Organic Design for Command and Control, a Discourse on Winning and Losing. Maxwell Air Force Base, Alabama Air University, Montgomery, AL, USA. Unpublished lecture notes, 1987.

5. Shewhart, W.A. *Statistical Method from the Viewpoint of Quality Control*; U.S. Department of Agriculture: Dover, UK, 1939; p. 45.

6. Kasser, J.; Mackley, T. Applying systems thinking and aligning it to systems engineering. *INCOSE Int. Symp.* **2008**, *18*, 1389–1405. [CrossRef]

7. Godfrey, P. What is Systems Thinking? How does it relate to Systems Engineering? INCOSE UK zGuide 7, Issue 1. March 2010. Available online: https://incoseonline.org.uk/Program_Files/Publications/zGuides_7.aspx?CatID=publications&SubCat=zGuides (accessed on 10 April 2018).

8. Pyster, A.; Olwell, D.; Hutchison, N.; Enck, S.; Anthony, J.; Henry, D.; Squires, A. (Eds.) *Guide to the Systems Engineering Body of Knowledge (SEBoK)*, version 1.0; The Trustees of the Stevens Institute of Technology: Hoboken, NJ, USA, 2012; pp. 98–125. Available online: http://www.sebokwiki.org (accessed on 29 March 2018).

9. INCOSE. *INCOSE, Systems Engineering Handbook*, 4th ed.; INCOSE-TP-2003-002-04-2015; Wiley: Hoboken, NJ, USA, 2015; pp. 18–21.

10. Burge, S.E. Systems Engineering: Using Systems Thinking to Design Better Aerospace Systems. In *Encyclopedia of Aerospace Engineering*; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2010.

11. Frank, M. Engineering Systems Thinking: Cognitive Competencies of Successful Systems Engineers. *Procedia Comput. Sci.* **2012**, *8*, 273–278. [CrossRef]

12. Billah, K.Y.; Scanlan, R.H. Resonance, Tacoma Narrows bridge failure, and undergraduate physics textbooks. *Am. J. Phys.* **1991**, *59*, 118–124. [CrossRef]

13. Petroski, H. *Design Paradigms—Case Histories of Error and Judgment in Engineering*; Cambridge University Press: New York, NY, USA, 1994; pp. 156–165.

14. Gaal, R. This Month in Physics History, November 7, 1940: Collapse of the Tacoma Narrows Bridge. *APS News*. November 2016, Volume 25. Number 10. Available online: https://www.aps.org/publications/apsnews/201611/physicshistory.cfm (accessed on 14 August 2018).

15. Dallard, P.; Fitzpatrick, A.J.; Flint, A.; le Bourva, S.; Low, A.; Smith, R.M.R.; Willford, M. The London Millennium Footbridge. *Struct. Eng.* **2001**, *79*, 17–35.

16. Josephson, B. Out of Step on the Bridge. *Guardian Letters*. 14 June 2000. Available online: https://www.theguardian.com/theguardian/2000/jun/14/guardianletters3 (accessed on 7 May 2018).

17. Strogatz, S.; Abrams, D.M.; McRobie, A.; Eckhardt, B.; Ott, E. Theoretical Mechanics: Crowd Synchrony on the Millennium Bridge. *Nature* **2005**, *438*, 43–44. [CrossRef] [PubMed]

18. Bachmann, H.; Ammann, W. *Vibrations in Structures Induced by Man and Machines*; Structural Engineering Document 3e; International Association for Bridge and Structural Engineering (IABSE): Zurich, Switzerland, 1987; Chapter 2: Man-Induced Vibrations, Appendix A: Case Reports.

19. Fujino, Y.; Pacheco, B.M.; Nakamura, S.-I.; Warnitchai, P. Synchronization of Human Walking Observed during Lateral Vibration of a Congested Pedestrian Bridge. *Earthq. Eng. Struct. Dyn.* **1993**, *22*, 741–758. [CrossRef]

20. Wernimont, E.; Ventura, M.; Garboden, G.; Mullens, P. *Past and Present Uses of Rocket Grade Hydrogen Peroxide*; General Kinetics, LLC: Aliso Viejo, CA, USA, 2018; p. 92656.

21. Duddu, P. The World's Deadliest Torpedoes. *Naval Technology*. 8 June 2014. Available online: https://www.naval-technology.com/features/featurethe-worlds-deadliest-torpedoes-4286162/ (accessed on 7 May 2018).

22. Guardian. What Really Happened to Russia's 'Unsinkable' Sub. *The Guardian*, 4 August 2001.

23. Faulconbridge, G. Nightmare at Sea. *Moscow Times*, 3 December 2004; Archived from the original on 28 February 2014.

24. Garfield, L. The 'Death Ray Hotel' Burning Las Vegas Visitors Came Up with a Simple Fix. *Business Insider*. 30 June 2016. Available online: http://www.businessinsider.com/the-vdara-death-ray-hotel-is-still-burning-people-in-las-vegas-2016-6 (accessed on 7 May 2018).

25. Maimon, A. Vdara Visitor: 'Death Ray' Scorched Hair. *Las Vegas Review-Journal*. 24 September 2010. Available online: https://www.reviewjournal.com/news/vdara-visitor-death-ray-scorched-hair/ (accessed on 7 May 2018).

26. Hodge, D. Reflective "Death Ray" Torments Vegas Sunbathers. *Reuters Environment*. 1 October 2010. Available online: https://www.reuters.com/article/us-lasvegas-deathray-idUSTRE6904AR20101001 (accessed on 7 May 2018).

27. Smith-Spark, L. Reflected Light from London Skyscraper Melts Car. *CNN*. 3 September 2013. Available online: http://edition.cnn.com/2013/09/03/world/europe/uk-london-building-melts-car/index.html (accessed on 7 May 2013).

28. BBC. Who, What, Why: How Does a Skyscraper Melt a Car? 3 September 2013. Available online: http://www.bbc.com/news/magazine-23944679 (accessed on 7 May 2018).

29. YouTube. Man Fries Egg in London Skyscraper Heat. *TheNetFrancesco*. 3 September 2013. Available online: https://www.youtube.com/watch?v=TGnZz8-PvdY (accessed on 7 May 2018).

30. Sherwin, A. Walkie Talkie City Skyscraper Renamed Walkie Scorchie after Beam of Light Melts Jaguar Car Parked Beneath It. *The Independent*. 3 September 2013. Available online: https://www.independent.co.uk/arts-entertainment/architecture/walkie-talkie-city-skyscraper-renamed-walkie-scorchie-after-beam-of-light-melts-jaguar-car-parked-8794970.html (accessed on 7 May 2018).

31. Washington Post News Service. London's 'Fryscraper' Draws Crowd on Hottest Day. *Mississauaga News*. 6 September 2013. Available online: https://www.mississauga.com/news-story/4067822-london-s-fryscraper-draws-crowd-on-hottest-day/ (accessed on 7 May 2018).

32. Rhee, J.; Ross, B.; Ferran, L.; Hosford, M. Toyota Recalls 2.17 Million Vehicles Over Pedal Entrapment. *ABC News*. 24 February 2011. Available online: https://abcnews.go.com/Blotter/toyota-recalls-21-million-vehicles-floor-mats-pedals/story?id=12989254 (accessed on 7 May 2018).

33. Evans, S.; MacKenzie, A. The Toyota Recall Crisis: A Chronology of How the World's Largest and Most Profitable Automaker Drove into a PR Disaster. *Motor Trend*. 27 January 2010. Available online: http://www.motortrend.com/news/toyota-recall-crisis/ (accessed on 7 May 2018).

34. Cramer, M. 11 Terrible Automotive Engineering Decisions. 2010. Available online: https://jalopnik.com/5570865/11-terrible-automotive-engineering-decisions/ (accessed on 17 August 2018).

35. Feldt, A.G. Chapter 2: General Systems Theory. 1986. Available online: http://www.holisticwisdom.org/hwpages/chapt%202%20-%20GST.htm (accessed on 18 June 2018).

36. Jon. R.I.P Microsoft Zune (2006–2011). *Methodshop.* 2011. Available online: https://www.methodshop.com/2011/03/microsoft-zune-rip.shtml (accessed on 13 August 2018).

37. Norman, D. Designing for People—Systems Thinking: A Product is more than the Product. *Interactions* **2009**, *16*, 52. [CrossRef]

38. Johns Hopkins Whiting School of Engineering. System Context Diagrams. 1 February 2014. Available online: https://ep.jhu.edu/about-us/news-and-media/systems-context-diagrams (accessed on 8 June 2018).

39. Monat, J.P.; Gannon, T.F. *Using Systems Thinking to Solve Real-World Problems*; College Publications: London, UK, 2017.

40. The Systems Thinker. *Guidelines for Drawing Causal Loop Diagrams*; Pegasus Communications: Waltham, MA, USA, February 2011; Volume 22, No. 1; Available online: https://thesystemsthinker.com/wp-content/uploads/pdfs/220109pk.pdf (accessed on 11 June 2018).

41. Kim, D.H. *Introduction to Systems Thinking*; Pegasus Communications: Waltham, MA, USA, 1999; pp. 5–10.

42. Federal Aviation Administration. *System Engineering Functional N2 Diagram*; Federal Aviation Administration: Washington, DC, USA, 2006.

43. Bechtel Jacobs Company. UF6 Cylinder Project Systems Engineering Management Plan. U.S. Dept. of Energy. July 1998. Available online: http://www.oakridge.doe.gov/duf6disposition/KTSO-017.pdf (accessed on 11 June 2018).

44. U.S. Department of Defense, Chief Information Officer. DoDAF Viewpoints and Models, Systems Viewpoint, SV-3: Systems-Systems Matrix. Available online: https://dodcio.defense.gov/Library/DoD-Architecture-Framework/dodaf20_sv3/ (accessed on 11 June 2018).

45. Wagenhals, L.; Levis, A. C4ISR Architectures and their Implementation Challenges, Lecture 4: A C4ISR Architecture Framework Implementation Process Case Study. INCOSE. Available online: http://slideplayer.com/slide/701808/ (accessed on 11 June 2018).

46. Hyatt, K. Roadshow's guide to car subscriptions, an alternative to buying or leasing. 14 June 2018. Available online: https://www.cnet.com/roadshow/news/new-car-subscription-service-guide-buying-leasing-2018/ (accessed on 27 June 2018).

47. Blanchard, B.; Fabrycky, W. *Systems Engineering and Analysis*, 5th ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2011.