2012

Learning from Failure in Systems Engineering: A Panel Discussion

Nathan Slegers
George Fox University, nslegers@georgefox.edu

Ronald T. Kadish

Gary E. Payton

John Thomas

Michael D. Griffin
University of Alabama - Huntsville

See next page for additional authors

Learning from Failure in Systems Engineering: A Panel Discussion

Nathan J. Slegers,[1,*] Ronald T. Kadish,[2] Gary E. Payton,[3] John Thomas,[4] Michael D. Griffin,[1,5] and Dan Dumbacher[6]

[1]*Mechanical and Aerospace Engineering, University of Alabama in Huntsville*, N274 *Technology Hall, Huntsville*, AL 35899
[2]*Booz Allen Hamilton*, 8283 *Greensboro Drive*, *McLean*, VA 22102
[3]SCI *Aerospace, Inc.*, 10140 *Community Lane*, *Fairfax Station*, VA 22039
[4]*Booz Allen Hamilton*, 13200 *Woodland Park Road*, *Herndon*, VA 20171
[5]*Center for System Studies, University of Alabama in Huntsville*, *Shelby Center* 136, *Huntsville*, AL 35899
[6]NASA *Marshall Space Flight Center*, MSFC, *Huntsville*, AL 35812

ABSTRACT

This paper summarizes the discussion of the Learning from Failure in Systems Engineering panel that was held in Huntsville, AL on November 8, 2010. The panel objective was to discuss how systems engineers respond to and learn from failure and identify future directions important to the community. The panel consisted of four representatives with experience in government, industry, and academia: (1) Ronald Kadish from Booz Allen Hamilton and former director of the Missile Defense Agency, (2) Gary Payton, retired Deputy Under Secretary of the Air Force for Space Programs, (3) John Thomas from Booz Allen Hamilton and President-elect of INCOSE, and (4) Michael Griffin from the University of Alabama, Huntsville and former NASA Administrator. Each panelist was asked to (i) provide an opening statement and elaborate on their experience with failure, (ii) describe when failure is appropriate, (iii) describe how we learn and react to failure, and (iv) identify and discuss techniques to improve how systems engineers react to failure. Several common themes arose from the discussion including: failure is an option, the importance of failure to allow reassessment, and more process is not the solution. Each of these is discussed in turn along with future directions identified for reacting to and learning from failure.

Key words: failure; product success; risk; lessons from failure; process

1. INTRODUCTION

The security and many aspects of a nation's economic prowess depends heavily on how well its aerospace, defense, and energy systems perform. Understandably, systems engineers spend a great deal of energy and resources avoiding the failure of these critical capabilities: Process and controls are instituted or improved; accountability reviews and milestones are defined; and historical experience is incorporated into organ-

*Author to whom all correspondence should be addressed (e-mail: slegers@mae.uah.edu; slegers@eng.uah.edu; kadish_ronald@bah.com; gary.payton@sci-aero.com; thomas_john@bah.com; mdg0007@uah.edu; daniel.l.dumbacher@nasa.gov).

izational culture and training [Hanawant and Rouse, 2010; Nagano, 2008]. Unintended consequences can arise, however, from the practice of avoiding and reacting to failure. Organizational culture can become risk-averse, thereby discouraging innovation and forgoing its rewards. An abundance of engineering process and controls can mask essential issues that affect a system's ability to meet strategic objectives. The pace of system design and development slows and costs rise with the encumbrance of reviews designed to eliminate, not just identify, and accept levels of risk. Historically, we have done a laudable job addressing the symptoms of a failure and implementing changes to eliminate or minimize reoccurrence [Sisson, 2005; Hessami, 1999]. However, we have not always done as thorough a job diagnosing the problem, creating the appropriate feedback loops to educate practitioners on the lessons, or creating the structural change required for a comprehensive and far-reaching solution.

It is typically recognized that failure is a common occurrence and that future success is often a consequence of our reaction to failure [Louthan, 2010]. Hazard analysis which relies on engineering expertise and judgment to identify, classify, and manage risk has continued to have an important role in foreseeing and preventing critical system failure [McKelvey, 1988; Collins, 2010]. Petroski [1985, 1994] has offered many thoughts on failure's role in engineering; including its value for design revision and failure as a source of engineering judgment. However, as the complexity of systems continues to increase, the difficulty of understanding systems and their failure also grows due to interactions and potentially unforeseen failures [Wears, Cook, and Perry, 2006]. Some have even acknowledged that it may not be possible for these complex systems to be designed with acceptable confidence according to our current expectations, and failure may be inevitable [Calvano and John, 2004]. The continued failure of important complex systems has lead to some asking: how did the system fail despite everything thought to be necessary in the way of process being done [Griffin, 2010]?

To gain a better understanding of failure's role in systems engineering and appropriate reactions to failure, a panel discussion on Learning from Failure in Systems Engineering was held in Huntsville, AL on November 8, 2010. The objective of the panel was to discuss how systems engineers respond to and learn from failure and identify future directions important to the community. The discussion was moderated by Dan Dumbacher, Engineering Director of the NASA Marshall Space Flight Center, and the panel consisted of: (1) Ronald Kadish from Booz Allen Hamilton and former director of the Missile Defense Agency, (2) Gary Payton, retired Deputy Under Secretary of the Air Force for Space Programs, (3) John Thomas from Booz Allen Hamilton and President-elect of INCOSE, and (4) Michael Griffin from the University of Alabama, Hunstville and former NASA Administrator. The discussion was structured around four questions with panelists asked to (i) provide an opening statement and elaborate on their experience with failure, (ii) describe when failure is appropriate, (iii) describe how we learn and react to failure, and (iv) identify and discuss techniques to improve how systems engineers react to failure.

The intention of this paper is to serve as a catalyst among the systems engineering community for broadening the discussion of how to learn from failure and its role. It is the authors' and panelists' hope that the subsequent summary will promote modifications and debate of the panelists' discussion and advance the state-of-practice in systems engineering. The following sections summarize the discussion that occurred during the panel. Section 2 addresses discussion point (i) and contains a brief overview of three specific examples of failure provided by the panel. Several common themes arose from the panel's examples and discussion of points (ii) and (iii) such as: Failure is an option, the importance of failure to allow reassessment, and more process is not the solution. Each of these themes is discussed individually in Sections 3, 4, and 5, respectively. Finally, the paper ends with a summary of the panel's closing remarks, suggestions for reacting to and learning from failure, and their implications to needs in changing the way we approach workforce development in systems engineering.

2. PANELIST EXAMPLES OF FAILURE

The panel provided a variety of examples where they experienced system failures at different stages and with varying levels of final success. These examples ranged from the X-33 reusable launch vehicle and Hubble Space Telescope to the Delta 180 powered space interceptor. A brief overview of each example follows.

2.1. X-33 Reusable Launch Vehicle

When it comes to systems engineering on a vehicle, Payton pointed to the X-33. The X-33 was a reusable launch vehicle program started in 1996 and was part of the broader Reusable Launch Vehicle technology demonstration program which also included the DC-XA, X-34, and X-37. The technical objectives of the X-33 program were to develop and demonstrate the use of lightweight composite materials for internal liquid hydrogen fuel tanks, linear aerospike rocket engines, a durable thermal protection system, and aircraftlike operations [US GAO, 1999]. Payton emphasized, as part of the program end goal, the teams intentionally reached for new technology on every subsystem of the X-33. Because it had been so many years since America had invested a significant amount in new launch vehicle technology, that type of catch-up game had to be endorsed, and it was fully suspected that there would be failures in ground tests. One of the many challenges was in the linear aerospike rocket engine, which had to use differential throttling for ascent attitude control. Due to its high risk, teams had alternative designs in their hip pocket in case that engine faced development problems. However, as in all programs, not enough resources, money, and facilities were available to have equivalent alternatives on every single subsystem of the X-33. Eventually one of the other subsystems experienced a technical failure in test. As Payton explained, based on the unprecedented nature of the program and its end goal, and under the conventional rhetoric that if you weren't occasionally failing, you weren't reaching far enough, that should have been embraced. The teams reached far, they struggled,

and they learned what didn't work. However, due to a lack of persistence and potentially a lack of constant vision at the agency level the program was canceled in 2001.

2.2. Hubble Space Telescope

One example provided by Griffin was his work on the Hubble Space Telescope, his first job as a systems engineer while at the Johns Hopkins University Applied Physics Laboratory. Griffin, along with a significant team of people, was asked to design and develop a backup system for the primary fine guidance sensor that would work on different principles [Griffin, Strikwerda, and Grant, 1984]. As part of the job, the team was asked to look across the Hubble of that day, not the Hubble that was eventually launched in 1990, and render a separate and impartial assessment of systems engineering on the telescope.

As chief engineer for that team, Griffin noted it was eye opening as problems were identified. It turned out that that the team was asked to assess the fine guidance system because the original design was based on a factor of 4 error in the number of photons that could be expected to be gathered in a given sample period. The result would be large errors in the visual magnitude of the stars they were trying to observe. In addition, it was found that people who were working the design of the solar arrays were not coordinating with people who worked the design of the control system. Therefore, as the solar arrays would swing in and out of the sunlight, they would irrevocably excite satellite motion in return and there was no image motion compensation or effective correction inside the control loop. Another problem was that the electronics were more appropriate to launch vehicle heritage than long-term satellites, with Hubble reaching its lifetime expected radiation dose in only a few months. Finally the team, along with many others, identified that there was no end-to-end test of the telescope.

Despite these errors, what stood out to Griffin was that each contractor could prove that they fully met the requirements of their side of the interface control document. Yet it was completely obvious, even to Griffin as a young systems engineer, that the device when fielded would not work. This proved to be true for the reasons described, as well as a variety of others. Certainly there were failures of system engineering in the effort of which the highly publicized flawed mirror was front and center. However, while being properly called a failure in systems engineering for several years, after the first servicing mission in 1993 the telescope became fully functional. The current result, due in large part to the persistence and commitment of the international scientific community, is that the Hubble Space Telescope has become a world icon.

2.3. Delta 180 Powered Space Interceptor

A second example provided by Griffin was his work as chief engineer on the first powered space interceptor, Delta 180 [Griffin and Rendine, 1988]. The Delta 180 was a success, and it performed a hit-to-kill intercept. While a success, one of the things that was not discovered until sometime afterward was that the endgame control simulations had not taken into account the hysteresis involved due to a small amount of stiction, a friction force that had to be initially overcome in the radar homing device. Overlooking the effect caused the control loop to lag just slightly from what was expected from the models and, as a result, produced a porpoising in the trajectory of just a few feet. As Griffin noted, fortunately, interceptor technology at the time was not good; therefore, the interceptor was very big. The porpoising error luckily was not large enough that it took it outside the diameter of the objects in question, and the system managed to score a hit-to-kill with only a glancing blow. If interceptor technology had been more advanced at the time, the interceptor most likely would have missed. However, because of its success a considerable time elapsed before anyone went back to look and found the ways in which the team had gotten lucky.

Overlooking the stiction in this case was not due to lack of attention to the endgame homing guidance problem. Three separate teams were simulating the homing guidance. The teams were allowed to talk to each other only under specific circumstances to ensure they were not corrupting each other with groupthink. Three independent simulations were desired, followed by technical interchange meetings. Launch occurred only when the collective circular error probables were within 0.5 m. Despite the process, nobody thought about the difference in the way the stiction was going to react in vacuum versus high altitude flight. The team was a bit lucky; in fact, three separate teams, as well as the systems engineer, missed something. An interesting lesson that could and should have been learned much earlier wasn't learned because of success.

3. FAILURE IS AN OPTION

Kadish started by declaring his shock at the panel title. He was shocked because, as he explained, most of the time people don't talk about failure; they talk about how to avoid failure. Society has been developing a hubris that we know how to do everything and that, every time we make a mistake, it is something we should have known better about or it was because of a poor management situation. There is a perception in society that demands success the first time and every time. In contrast to perception, it is not possible to always avoid failure; therefore, it is vital to think and talk about what it means to learn from failure.

The hubris Kadish described is also apparent from the ubiquitous use of the phrase "failure is not an option" made famous by the 1995 film Apollo 13. Gene Kranz, Flight Director during the Apollo 13 space mission, later wrote a book by the same name because he felt it reflected the attitude of Mission Control [Cass, 2005]. Subsequently "failure is not an option" has been used by many as a rallying cry, often naively attempting to declare that if they decide to take failure away as an option, they must necessarily succeed. A source of such confusion may be ambiguity in the definition of failure. Kadish pointed out that he has a problem with the definition of "failure" in systems engineering. According to Kadish, what failure means is you don't accomplish what you set out to as an end item, what we regard as being necessary in the end. From his experience, failure is an option at every step except the final goal. Thomas concurred with the assess-

ment and embraced the idea that failure is critical in the intermediate steps leading to the end objective. Returning to the X-33 example, because teams reached for new technology on every subsystem, failure at the intermediate steps was fully suspected. As Payton emphasized, those intermediate failures should have been embraced because of its value in pushing new technology.

"Failure is not an option," used by Kranz in his book to describe Mission Control's attitude, is more consistent with the panel's definition of failure, than how it is often used. Clearly, failure was an option as demonstrated by the oxygen tank explosion which crippled the command module and initiated the tense events to follow. In addition, further intermediate failures were possible as the crew, flight controllers, and support personnel engineered a safe return. Indeed as the panel stated, failure is an option at every step except the final goal, which in the case of Apollo 13 was a safe return of the crew.

Evaluation of the three examples provided by the panel also demonstrates that failure is an option. In each example, failures occurred or easily could have occurred at intermediate steps. In X-33, failure occurred on subsystems that, due to finite resources, did not have an alternative technical strategy. Multiple failures occurred on the Hubble Space Telescope, some identified prior to launch, others after. Failure was also an option, though narrowly avoided on the Delta 180 because, as Griffin put it, they were a bit lucky.

Abundant failures occurred at intermediate steps in both X-33 and Hubble. Despite failure being present, only X-33 resulted in failure of its end goal. The lessons to be learned lie in how failures were treated differently in the two cases. Griffin explained that in Hubble there certainly were failures of system engineering. In the end, who knows about these failures other than those who were intimately involved with the Hubble? In the end it has become an icon, not only an American icon but a world icon. Its pictures adorn art museums as well as computer screens of astronomers. What accounted for that was persistence, the willingness to learn from failure, to not accept final failure, to apply the necessary corrections in the wake of intermediate failures on the way to success, and in the end, to do what was necessary to achieve success. If someone tells you today that the Hubble overran by a factor of 3 or more, is there anybody who actually cares? Probably not, because of what it has done and what it has become. Those are some of the lessons to learn from what could have, for several years, only properly been called a failure of system engineering. That is not what we call it today.

In contrast to the persistence of Hubble stands X-33. As Payton described, initial commitments existed. The rhetoric of the day was if you weren't having the occasional failure, you weren't pushing hard enough; you weren't reaching far enough. That is great rhetoric, but at the first instance of failure, all that rhetoric often evaporates. You must have persistence in your mission; a dedication to your end goal at the agency level. Simple rhetoric is not adequate to deliver the success that is needed. The need for persistence in the face of failure has another dimension which was identified by Kadish. To be persistent, you need to set expectations right. There are things other than just the technical problem that applies. You not only learn technical lessons from failure, you learn that getting the needed resources and setting the political tempo as best you can is also important.

4. FAILURE PROVIDES THE CHANCE TO REASSESS

The panel embraced the notion that failure is an option and furthermore, for a system, it should be expected that failure at an intermediate stage will occur. Since we cannot absolutely avoid failure, what should its role be, and what does it mean to learn from failure? Thomas proposed that failure is the thing that provides the chance to reassess, to reassess how well we are listening to the team, to reassess how accurate and at what level of voracity are the assumptions that we are making, to reassess the viewpoint of our perspective of the problem that we have defined.

The insight that failure's value may be the role it plays for reassessment was extended beyond engineering by Griffin. He identified Bobby Jones as a golfer he greatly admired and noted Jones once said "I never learned anything from a match I won." Griffin connected this back to his experience on Delta 180 in Section 2.3, where because they were successful, they were denied the opportunity to learn from reassessment until much later. When talking about successful tests, we tend not to study them if the test went as expected. Generally there is buried in any success many instances of, "Wow, we dodged a bullet on that and we just didn't know it." As with many series of tests, more can be learned from a single failure than can be learned from all the successes. As Kadish clarified, this is often the case because there is hubris on our part and we don't look as critically at successes as we should when compared to the case when failure forces us to reassess. These thoughts are also reinforced by Petroski's insight that failure has the ability to nurture humility and caution [Petroski, 2007].

Reassessment as the opportunity birthed from failure can be related to the identity of a systems engineer. Thomas put forward this hypothesis: Can system failures be attributed to the system engineer who has lost their identity as they have that chance to reassess? Since failure is an option at intermediate steps, the systems engineer has a responsibility to reassess, a responsibility to learn, and a responsibility to make decisions in the face of failure. When systems fail in their end goal, is it because of a failure at an intermediate stage or failure of the system engineer to seize the chance to reassess when the intermediate failure occurs?

5. MORE PROCESS IS NOT THE SOLUTION

5.1. What Process Can't Do and Why It Is Not a Solution

One observation of the panel was that failures of system engineering process in the past typically resulted in the addition of more process. Aversion to all failure has resulted in substitution of abundant processes, analysis, and band aids to prevent us from failing at every step along the way. Systems engineers have often substituted that for the real creative process, leadership, and discipline in accomplishing what was

set out to do. The result is an encumbrance of rules, regulations, policies, and laws for a system which should be much more fluid. Griffin pointed out that if you didn't have process and someone said "you had a failure of process," then you probably should attend to that. But after 50 years of process, and with the layers and layers of what can only be described as bureaucratic process, he has become jaded as to the marginal value of more process. If systems engineers are to achieve success through failure, the community must be prepared to do something different. Process is not an answer to failure. It might be a part of the answer, but is not the answer to a technical setback. As leaders there are other options.

To be clear, the panel is not suggesting that the methods, processes, and tools which characterize systems engineering today are entirely defective but, rather, the misuse of process in replacement of thinking and self-accountability is amiss. Similarly, the panel is not suggesting that process is the source of the problem, but rather misuse of process to address a failure that process cannot solve is the problem. An example of the misuse of process is the addition of an extra layer of review in response to failure, when the real problem was not the existing process, but rather that the original reviews were incompetent. Processes can be wrong; they can have errors or failures themselves. As highlighted in the previous example, many times the best starting point is correction of existing errors, not defining new processes.

In order to identify when process is an inappropriate reaction to failure the panel identified five tasks that simply adding more process can't do. Kadish identified the first two from his experiences where people have proven without a shadow of a doubt that they followed every step of the process yet still failed. In these situations one needs to ask, "Would more process help?" The reason for failure is often not the process, but that the team didn't understand what they were doing in the first place; and the need is to find the right type of leadership with an understanding of how the system is supposed to work. In such a situation more process is not a solution because (i) process can't replace leadership or understanding and (ii) good process well followed does not prevent failure. Both are demonstrated by the importance of human factors in failure with some case studies provided by McEvily [2004].

Griffin, an avid pilot, used his preflight checklist to illustrate other items process can't do. His checklist is unique to his airplane and that checklist is "process," nothing but process. However, passing the checklist does not do anything to teach you how to fly. As with flying, and similarly in engineering, there is more than one skill involved. Engineers need to have process standards by which we execute our day-to-day chores; it saves us from making an error of omission, just as having a checklist saves a pilot from taking off with the flaps down. However, just as with the pilot's checklist, process has limits to what it can do: (iii) Process does not make a bad design better, (iv) process does not help distinguish a good design from a bad design, and (v) process doesn't tell you what to do if you have identified a bad design. Process does not do anything at all with regard to very significant pieces of the overall engineering profession. If we ever allow ourselves, as individuals or as organizations, to believe that those other pieces of the engineering development profession are not equally important, then we will fail; we will just fail for another reason. In engineering, like flying, you need to follow the checklist, but you also need to know how to engineer.

5.2. Why We Fall Back on Process

Despite the previously mentioned limitations of process to provide solutions to failure, leadership, managers, and engineers typically respond by the addition of more process. An important step in learning from failure and determining what needs to been done differently is understanding why process is the natural human reaction of the leadership.

In a position where you suffer the scars of a major systems failure, where programs can get cancelled or other deleterious events happen, you have to struggle with the idea of accountability and the issue of leadership. What do you do when you have a large organization that has focused on a program and something goes wrong: You have failure; you have a setback? Or you have a few setbacks that are not expected. What does the leadership do to react to that? Kadish submitted that, having been in that situation, you wring your hands and say "If I put more process in, I can fix this." Absent another suggestion, that tends to be what happens. Human nature feels that if they levy more process in the wake of any problem, they can point to that as something they did. Griffin points out that other options exist. However, one is less able to demonstrate as a manager that you have done something if you replace personnel at the top who didn't have a grasp of the system with another person whom you feel has a better grasp. When you have to defend that decision in public or in the bureaucracy people ask, "Can you prove it?" Of course you can't "prove it," and that makes the latter option uncomfortable, even though it may be more likely to be a productive solution than simply more process.

Thomas put an edge on the discussion by framing it another way. When you are dealing with a bureaucracy, process is to a bureaucracy as heroin is to an addict; it is the thing that makes you feel safe, and it is the thing that allows you to assume something else is taking care of the problem. More process offers the illusion of control. It is in fact just an illusion, but it does exist as an illusion. It is a case where to recognize anarchy is ineffective; one then concludes a situation under complete control must therefore be the most effective. As with most things in life, the best result is somewhere in the middle. It was the writer Madeleine L'Engle who wrote, "When we were children, we used to think that when we were grown-up we would no longer be vulnerable. But to grow up is to accept vulnerability...." [L'Engle, 1980, p. 145]. Similarly, for young systems engineers, it is often expected that wielding process can remove vulnerability to failure. It is only as they mature that they begin to accept vulnerability to failure, how to learn from it, and how to harness it to achieve success through failure.

5.3. Consequences of Excessive Process

While simply adding process is likely not the right answer in response to any particular failure, indulging people's addiction to process may seem harmless as long as the additional process is innocuous. However, if additional layers of unnec-

essary process do indeed have malicious consequences, their addition simply to satisfy human desires or remove leadership from the hot seat of public opinion may be severely detrimental. As the panel continued discussion of the role of process as a response to failure, two consequences of excessive process were identified.

The first consequence was offered by Thomas. He submitted that while process is necessary, human behavior seems to evolve so that process, when not watched very carefully, seems to remove self-accountability, self-thinking, and retrospection. In all successful projects there is a level of personal commitment, integrity, and accountability, whether it is among engineers or technicians. The more process you lay on top, the more layers of inspectors, the farther away you get from that personal level of accountability. Empowerment is as important a factor as accountability. Empowerment doesn't replace it, but being able to empower the team so that they can tap into the sources of that human identity component is critical in moving forward and tapping into innovation, finding a solution that people never thought about before. Excessive process also happens to extract the sense of empowerment away from the team because of lack of control. In these cases people feel power is in the process, not in the individual.

The second consequence of excessive process was identified by Griffin in what he called "an underlying truth about resources." The underlying truth is as follows: There is only so much effort and capability to go around. It's like funds in a bank; there is only so much attention you can put on any design. You have a quantity of people times their available time, and that is the total effort you, as a manager, can allow to be expended on a project. Effort must be allocated on those things which matter and prevented from being utilized on those things which don't. Of course, the wisdom lies in knowing the difference. Good systems engineers are those who have a highly developed executive function. The ability to constrain themselves and to constrain their teams from indulging in expending effort where it is not appropriate, the ability to husband that bank of resources you have and spend it on the right things is what makes the difference between success and failure. You have to choose; success lies in choosing wisely, but not choosing at all is almost a guarantee of failure. Engineers are not always so good at that yet. Some people know how to do it and some don't. No individual and no team can work all of the possible details that need to be worked, but that fact is not an excuse to hide behind process. You can fail for many reasons. You can fail because you broke a piece of hardware, or your program can fail because you ran it so far behind schedule and so far over budget that somebody cancels it for you. We don't think enough about that mode of failure. That mode of failure is facilitated by adding layers of excessive process.

5.4. Role of Process in Failure

Previously five tasks process cannot do were identified by the panel along with consequences of excessive process. However, obvious to all system engineers is that, over the last 50 years, methods, processes, and tools have been developed by the discipline which have advanced the art of design. To be successful in learning from failure, it is necessary to also discuss the role of process in failure, in addition to just identifying what is not its role.

Thomas connected the role of process in failure to his earlier hypothesis about the identity of systems engineers. He submitted identity comes from the attitude of oneself, a viewpoint of your role in the program and a perspective of how you fit into the team. I would challenge us to reflect back over the last 50 years. Has that identity and self-identity, by the individual or the larger organization, evolved from the system engineer who is a visionary and technical leader of people, process, and tools, or a systems engineer who is the steward of process? Systems engineers to be admired go beyond the category of craftsmen. Craftsmen are people who understand process; they are people who understand methods, techniques, and tools. System engineers, however, are much more than just craftsmen; they have deep technical disciplines, they have broad abilities to go across multidisciplinary fields, and they have environmental knowledge that gives perspective across the domains whether it crosses the missions and technologies. Systems engineers to be admired have enormous problem solving skills; they have critical thinking skills, systems thinking skills, associative thinking skills that show a curiosity of events and the analysis of causality. The systems engineers to be admired have extraordinary leadership and team-building skills, communication, and conflict management. It is those skills that will help avoid ultimate failure more times than not, rather than simply process itself.

Process utilized by a system engineer with proper identity as described above has a clear role in failure. Process, when wielded by the previously described systems engineer, provides efficiencies and the ability to channel the work of hundreds or thousands. But process by itself is a dangerous two-edged sword. Process alone is not an answer to failure; it may be a part of the answer. As leaders there are other options; process is just one.

6. CLOSING REMARKS

From the beginning the panel emphasized the need to talk about learning from failure in our larger society so that we can be successful in the end state, while tolerating the inevitable failure along the way. It was asked, how do we get back to that leadership and discipline that our ancestors used to get us where we are today and not fall back on bureaucracy and process only to make it work, or in the worst case just to survive? It was Kadish's insight into the mischaracterization of failure that set the stage. He submitted failure means you don't accomplish what you set out to as an end item; also said, failure is an option at every step except the final goal. This notion of failure was proposed in contrast to the common perception that failure must be avoided at every step along the way. The idea that failure is critical in the intermediate steps leading to the end objective was embraced by the panel.

While failure at intermediate stages is often supported in rhetoric, in the face of failure the rhetoric evaporates, and the solution is often more process. It is this levying of more layers of bureaucracy as a natural reaction—"process is to a bureaucracy as heroin is to an addict" as Thomas put it—which serves

only to allow leadership to feel like they did something. Unfortunately, improper wielding of process in response to failure also serves to diffuse the value of failure, diffuse the ability to learn from failure, and diffuse the chance to be successful through failure. Improper wielding of process in response to failure masks the opportunities available in failure. Namely, that failure provides the chance to reassess.

Some of the lessons learned put forth by the panel came directly from their reflection on involvement in past programs and comparison of what were identified as "successful programs" with those that failed. One lesson learned from failure was identified by both Payton and Griffin as they describe X-33 and the Hubble Space telescope, respectively. In both cases, failures of system engineering occurred; however, while X-33 was terminated for intermediate setbacks, Hubble continued in spite of setbacks and in the end became an icon. What accounted for that was persistence, the willingness to learn from failure and, in the end, do what was necessary to achieve success.

Near the end of the discussion Payton drew parallels over several projects. The projects included Intercontinental Ballistic Missile programs; Atlas, Titan, Minuteman I, II, and III, Apollo with its presidentially dictated schedule, and early Strategic Defense Initiative Organization projects. In all those projects, the schedule pressures and pace forced individual accountability because there wasn't time to have multiple teams scrutinize every decision or have research council studies to determine what you were going to go do. A thread through all those successful programs has been a schedule driver that forces individual accountability that expedites the decision making, saves money, and gets the job done. The panel strongly championed the notion of schedule pressure, if for no other value than it forces decisions. As Griffin added, it is better to make a wrong decision and find that out by nature than to study forever trying to make sure you don't make a mistake. The lesson learned put forth by Payton was that leaders have other options available to them other than more process. In the specific example laid out, the schedule promoted accountability and forced decision making in contrast to more process which happens to extract empowerment and remove accountability.

Emphasized throughout the discussion was the lesson that good process well followed does not prevent failure. Process is not the problem, but rather the misuse of process in replacement of thinking and self-accountability. This highlights the concept of leadership, discipline, accountability, and understanding what one is trying to do. Processes are just a part of that. If there is a path forward, it lies in rethinking what we mean by the system engineer who does systems engineering and rethinking how we train the systems engineer. As Thomas earlier explained, successful systems engineers have many skills; enormous problem-solving skills, critical thinking skills, and associative thinking skills, to name a few. Those skills will result in ultimate success more times than not rather than simply process itself. Systems engineers need to think of themselves as being in the design business rather than stewards of process. They need to own and be responsible for design, not the process. A necessary part of training systems engineers who can learn from failure and achieve success through failure is the realization that only through experience can one develop the scars of decision making, and if one is not building things it is hard to develop experience. A final thought was added by Griffin where he emphasized to the community that in training systems engineers we need to be training chefs not cooks, Chief Financial Officers not accountants. If we think of it along those lines, success can lie at the end of the path.

REFERENCES

C.N. Calvano and P. John, Systems engineering in an age of complexity, Syst Eng 7(1) (2004), 25–34.

S. Cass, Apollo 13, we have a solution: Part 2, IEEE Spectrum Mag (April 2005).

R.L. Collins, Process hazard analysis quality, Process Safety Progress 29(2) (2010), 113–117.

M.D. Griffin, How do we fix systems engineering, 61st Int Astronaut Cong, Prague, Czech Republic, September 27–October 1, 2010.

M.D. Griffin and M.J. Rendine, Delta 180/vector sum: The first powered space intercept, 26th AIAA Aerospace Sci Meet, Reno, NV, January 11–14, 1988.

M.D. Griffin, T.E. Strikwerda, and D.G Grant, The space telescope alternate fine guidance sensor, Guidance and Control Conference, Seattle, WA, August 20–22, 1984, pp. 143–153.

E.S. Hanawalt and W.B. Rouse, Car wars: Factors underlying the success or failure of new car programs, Syst Eng 13(4) (2010), 389–404.

A.G. Hessami, Risk management: A systems paradigm, Syst Eng 3 (1999), 156–167.

G. Kranz, Failure is not an option: Mission control from Mercury to Apollo 13 and beyond, Simon & Schuster, 2009.

M. L'Engle, Walking on water: Reflections on faith and art, H. Shaw, Wheaton, IL, 1980.

M.R. Louthan, Overcoming failure, J Fail Anal Prevent 10(4) (2010), 249–250.

A.J. McEvily, Failures in inspection procedures: Case studies, Eng Fail Anal 11(2) (2004), 167–176.

T.C. McKelvey, How to improve the effectiveness of hazard and operability analysis, IEEE Trans Reliab 37(2) (1998), 167–170.

S. Nagano, Space systems verification program and management process, Syst Eng 11(1) (2008), 27–38.

H. Petroski, To engineer is human: The role of failure in successful design, St. Martin's Press, New York, 1985.

H. Petroski, Design paradigms: Case histories of error and judgment in engineering, Cambridge University Press, Cambridge, 1994, pp. 121–143.

H. Petroski, The paradox of failure, Los Angeles Times (August 4, 2007), A17.

R.D. Sisson, Failure analysis and the need for enhancing communications, J Fail Anal Prevent 5(1) (2005), 18–23.

US GAO, Status of the X-33 Reusable Launch Vehicle Program, GAO/NSIAD-99-176, US General Accounting Office, Washington, DC, 1999.

R.L. Wears, R.I. Cook, and S.J Perry, Automation, interaction, complexity, and failure: A case study, Reliab Eng Syst Saf 91(12) (2006), 1494–1501.

Nathan Slegers received a B.S. in mechanical engineering from the University of Washington, Seattle, Washington, in 2000. He received a Master's Degree in 2002 and a Ph.D. in 2004, both in mechanical engineering, from Oregon State University, Corvallis, OR. In 2005, he joined the faculty of the Department of Mechanical and Aerospace Engineering at the University of Alabama in Huntsville and is currently an Associate Professor. His research involves dynamic modeling and control systems engineering with application to flight mechanics, smart weapons, precision airdrop, and unmanned systems. Particular emphasis is in developing unique mathematical models and control for a variety of complex systems.

Ron Kadish is Senior Vice President at Booz Allen Hamilton and former Director of the Missile Defense Agency. Ron retired from the US Air Force as a Lieutenant General in 2004 with 34 years of active service. As a Senior Pilot, Ron logged more than 2500 flying hours, primarily in the C-130. His last assignment was as Director, Missile Defense Agency, where he served as the acquisition executive for all ballistic missile defense systems and programs. Ron formerly commanded the Air Force Electronic Systems Center, the Air Force's Center of Excellence for command and control systems. He served as the Program Director for the F-15, F-16, and C-17 program offices and as the Director for Manufacturing and Quality Assurance for the B-1 System Program Office. Ron led the congressionally mandated Defense Acquisition Performance Assessment (DAPA) Study due to his extensive experience in systems acquisition and program management. Today, Ron is the Senior Vice President serving as the Capabilities Development Officer for Booz Allen Hamilton's Center of Excellence for Acquisition and Program Management.

Gary Payton is retired Deputy Under Secretary of the Air Force for Space Programs. In this capacity he oversaw the development and procurement of all military space capabilities. These include constellations of navigation, weather, communication, and surveillance spacecraft, plus space launch systems. Prior to returning to the Air Force, Mr. Payton was the Deputy for Advanced Systems in the Missile Defense Agency (MDA). There he led the MDA technology program to deliver future ballistic missile defense sensors, weapons, and battle management capabilities. Before joining MDA, Mr. Payton was the Senior VP of Engineering and Operations for Orbimage, now renamed GEO-Eye. Prior to joining Orbimage, he was the Deputy Associate Administrator for Space Transportation Technology at NASA/HQ. There, he initiated, planned, and led the Reusable Launch Vehicle technology demonstration program, which included the X-33, X-34, X-37, and DC-XA flight test projects. Mr. Payton retired as an Air Force Colonel with over 23 years of service where he served as a pilot, instructor pilot, spacecraft operations director, and space technology manager. In 1985 Colonel Payton flew as a Payload Specialist onboard the Space Shuttle Discovery in the first military flight of the Space Shuttle program.

John Thomas is Senior Vice President at Booz Allen Hamilton, as well as its Lead Systems Engineer, and the President-elect of the International Council of Systems Engineering (INCOSE). He leads teams that plan and execute multimillion dollar complex system programs. His clients include U.S. government and industries worldwide. John's current focus is supporting complex systems development activities for the U.S. federal government. With John's Systems Engineering and Integration (SE&I) stewardship, Booz Allen (a $5 billion strategy and technology consulting firm) has invested in its Systems Engineers and SE methodologies for the purpose of enhancing the SE profession and its practitioners. John's assignments span leadership roles encompassing both programmatic and engineering activities. He is experienced in conducting strategic planning, communications strategy development, conflict management, business process analysis and leading decision making processes.

Michael Griffin is UAHuntsville's King-McDonald Eminent Scholar and Professor of Mechanical & Aerospace Engineering, Director of the Center for System Studies, and former NASA Administrator. Prior to rejoining NASA, he was Space Department Head at the Johns Hopkins University Applied Physics Laboratory. He has also held numerous executive positions with industry, including President and Chief Operating Officer of In-Q-Tel, Chief Executive Officer of Magellan Systems, General Manager of Orbital Science Corporation's Space Systems Group, and Executive Vice President and Chief Technical Officer at Orbital. Mike's earlier career includes government service as both Chief Engineer and Associate Administrator for Exploration at NASA, and as the Deputy for Technology at the Strategic Defense Initiative Organization. Prior to joining SDIO in an executive capacity, he played a key role in conceiving and directing several "first of a kind" space tests in support of strategic defense research, development, and flight testing. These included the first space-to-space intercept of a ballistic missile in powered flight, the first broad-spectrum space-borne reconnaissance of targets and decoys in midcourse flight, and the first space-to-ground reconnaissance of ballistic missiles during the boost phase. He also played a leading role in other space missions in earlier work at the JHU Applied Physics Laboratory, NASA's Jet Propulsion Laboratory, and the Computer Science Corporation. Mike previously taught for 13 years as an adjunct professor at the University of Maryland, the Johns Hopkins University, and George Washington University.

Dan Dumbacher is head of the Engineering Directorate at NASA's Marshall Space Flight Center in Huntsville, AL. Prior to 2007, Dumbacher served as deputy manager of the Exploration Launch Projects Office, where he assisted in the overall project management of NASA's Ares I crew launch vehicle and Ares V cargo launch vehicle. Dumbacher has also served as deputy director for product assurance in the Safety and Mission Assurance Office at Marshall, focusing on efforts to return the space shuttle to flight. Dumbacher joined NASA in 1979 at the Marshall Center and from 1994 to 2004 served in a variety of Marshall leadership positions related to advanced space transportation research and technology development. His positions included: manager of Marshall's X-37 Flight Demonstrator Project Office, manager of the Delta Clipper-Experimental Advanced Flight Vehicle Project, deputy manager of the X-33 Program, and deputy manager of the Orbital Space Plane Program. Dumbacher, an Indianapolis native, received a bachelor's degree in mechanical engineering from Purdue University in 1981. He received his master's degree in administrative science from the University of Alabama in Huntsville in 1984 and completed the Senior Managers in Government study program in 2002 at Harvard University in Cambridge, MA.