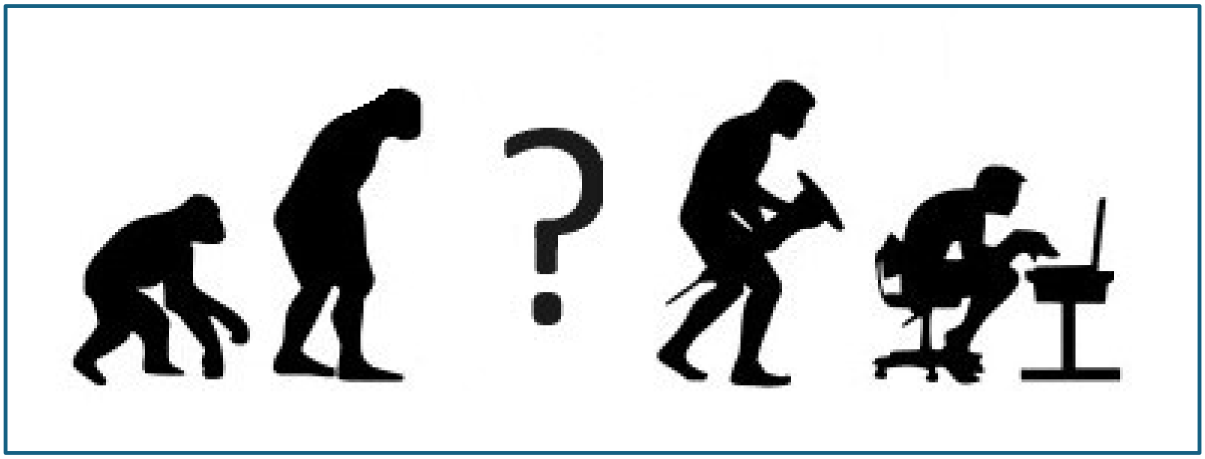
The Missing Links of System Safety



Chapter 1 - The Risk Managing Authority

In this new series of posts, I want to focus on specific areas of MIL-STD-882 that are not always part of the discussions on the current industry safety challenges but I believe are essential elements.

Each area represents a vital link in the system safety process consistent with lessons learned from the ICBM program. From my experience, if any of these links are missing or malformed, it can undermine a system safety effort.

The first is what I refer to as the risk managing authority (RMA).

MIL-STD-882 mentions a "risk acceptance authority" a number of times while making other references to some entity carrying authorizing responsibilities within the safety process. I lump these tasks together as the RMA. This role is the lynchpin of the system safety effort.

The RMA should be a trained (preferably independently certified) and experienced safety professional. The safety "buck stops here" individual for each product with a clear vision for the system safety process to be conducted.

A RMA should be identified for every level of the product procurement chain; from the regulator to the purchaser down to all prime and sub-contractors and suppliers.

Each level RMA has ownership of the formulation and execution of the safety plan for their "piece of the pie". They should own all risk characterization and assure coordination of assessments for all integration safety issues between the other pertinent product RMAs.

By far the most crucial assignment is the RMA for the procuring organization. They provide high level safety requirements and metrics tailored to the specific product and end use. They determine the applicable safety specifications and standards that will be used and coordinate with the regulatory agency as needed throughout the development cycle. The purchaser RMA is also the "final word" on safety issues for all product RMAs and represent the end user community in residual risk assessments.

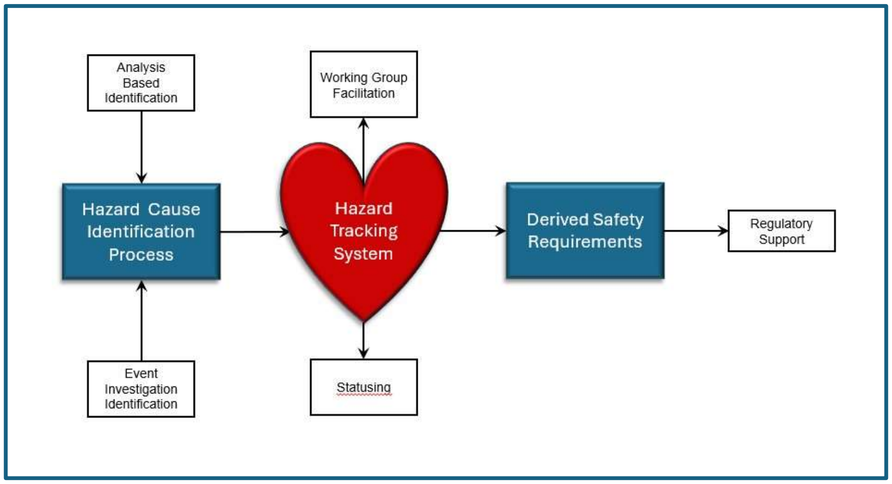
Failure to put a suitably qualified individual in this position can lead to a wayward safety process. As a minimum, it carries the potential for needless churn and inefficiencies in the safety process. If this role is added or changed late in the game, "catch up safety" can be many times harder even for a highly trained and experienced safety practitioner.

To satisfy independence learned from the ICBM era, the RMA should work closely with but be independent of the engineering/product organization.

My observation across the industry is that the right people in an RMA position can galvanize the safety effort and make it efficient and effective, while a poor selection of RMA has the potential to relegate the safety effort to an inefficient or ineffective exercise.

Chapter 2 - The Hazard Tracking System (HTS)

The next "link" isn't missing so much as I feel it is currently not given enough industry focus. Maybe more of a weak link. I consider the HTS, as defined in MIL-STD-882 Task 106, as the heart of the system safety effort. The Working Group (Task 105) provides the heartbeat (to maintain the analogy) and is the subject of a future post.


Analysis Based Identification
Working Group Facilitation
Hazard Cause Identification Process
Hazard Tracking System
Derived Safety Requirements
Regulatory Support
Event Investigation Identification
Statusing

My observation is that the HTS is primarily implemented for its role in facilitating closed loop lifecycle tracking of safety data. Besides making sure identified safety issues are run through the process to closure, it can provide assurance that nothing "falls through the cracks", particularly during the potentially chaotic run-up to regulator approval. It can be as simple as a spreadsheet but if done right, can provide a leveraging point for integrating with other processes (see below) as well as facilitating statusing and reporting of the safety effort throughout the product development team (including leadership).

Full disclosure is that my previous IT career experience as an Oracle DBA allowed me to design and develop a customized relational database version of a HTS during my system safety tenure. It's not the greatest since I was busy understanding MIL-STD-882 and running multiple complex product development safety programs at the same time. But it worked and is still being used to run a number of advanced product development efforts. I know it can be done right.

The COTS tools I examined before creating the custom solution were directed at supporting sustainment safety efforts. Primarily field reporting and corrective action system (or FRACAS for short) which were focused on managing incident response and fleet risk. These tools do well for managing risk after finding what has gone wrong, but are not suited for the development environment where the effort is around what can potentially go wrong.

The emergence of the model based systems engineering (MBSE) approach is highlighting the need not only for the existence of a common HTS tool functionality, but also a level of integration with other development processes and software tools. The HTS is the key link to the safety requirements in the MBSE paradigm.

If properly implemented, the HTS is the key to an efficient and focused safety effort and a "force multiplier" for the product safety team. Trying to conduct the safety data management and reporting task during development manually is an exercise in tedium and inefficiencies with many opportunities for "drops" and oversights.

In order to face the many safety challenges facing the industry today, safety process needs the right tools in place with the HTS at it's heart.

I welcome any comments or information on existing COTS HTS tools you know about which support the product development safety process.

Chapter 3 - The Safety Working Group (WG)

Like the Hazard Tracking System (HTS), the safety working group (MIL-STD-882 Task 105) isn't so much a missing link as it is an often-misapplied link in the system safety process. Also, the HTS (see last week's post) and WG are linked in such a way that a weakness in one can compromise the function of the other.

Using an analogy from my previous post, if the HTS is the heart of a system safety effort, then the flow of safety data during the product lifecycle is regulated by the beat of the WG. WG's should have a regular meeting rhythm in order to keep the safety process healthy and vital.

The central purpose of any WG is to process safety data to closure. It's where the core safety work gets done. Sleeves are rolled up, hazards and causes are discussed with the right people engaged and corrective actions identified and evaluated. Focused and open safety discussion should be encouraged and details worked out. WG's exists on multiple levels depending on the organizations' safety policy and where the project is on the product lifecycle.

During operational programs, WG's are about taking action and during new product development, they are about reaching consensus. Trouble can come if the same WG model is used for both phases of the product lifecycle. Since development programs typically process a larger amount of safety data concurrently, the development WG structure needs an additional "layer" at the individual IPT (Integrated Product Team - e.g., engineering teams) level where the WG focus is processing safety data pertinent to that IPT. It is here that their time and expertise can be more efficiently leveraged than a WG at the product level.

Having a well-functioning HTS facilitates this review by serving each IPT their "piece of the safety pie" to work on at each WG. Repeating regular IPT WG's until all data has been processed and SME concurrence is reached by the time the design is frozen.

A couple of perspectives I have tried to impress on teams is that a WG is distinct from a review. A WG isn't about judging data right or wrong as it is about reaching consensus. Prior to regulatory approval, every safety meeting is a working meeting. Any new safety data found before the system is fielded is goodness.

Under the guidance and facilitation of a trained and experienced professional safety team, the WG is the assurance to the entire team as well as future operators and users that the hard work of safety has been done. All the safety data has been fully vetted at the right level and residual risk is understood by the time regulatory approval is sought.

My message to the IPT's is that WG's aren't just about ultimately convincing a regulator that the product is safe as much as it is about convincing ourselves first.

Chapter 4 - Risk Acceptance Matrix (RAM)

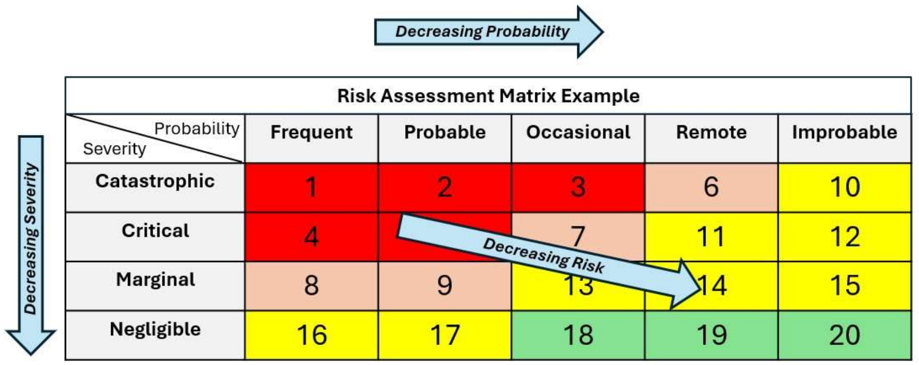
Based on my experiences as a safety professional and safety certification instructor as well as reading LinkedIn posts and comments, this topic has the potential for strong reaction from my fellow safety professionals. So let me start with the caveat "from my interpretation" to give room for their comments.

Therefore, I'm not presenting the RAM as much of a missing link as a regularly misinterpreted link.

The RAM (MIL-STD-882 Table III) fundamentally provides product risk acceptance guidance. It provides a metric for risk management of the whole product lifecycle and is intended to be tailored to the product. Further, it can evolve as a product passes through its lifecycle phases.

It's the responsibility of the procuring Risk Managing Authority (RMA - see my previous post in this series) to define the product RAM and get key product stakeholder and regulator concurrence at the start of a product development effort as well as at any point in the product lifecycle where subsequent regulatory guidance is needed. This guidance is to be passed down to contractors and suppliers. Typically in the SOW.

The RAM is relatively straightforward. A "high to low" matrix is created featuring definitions for levels of failure/malfunction outcome severities with associated probability thresholds. Then, zones of equivalent risk (severity x probability) are defined on top of the matrix. Each of these zones are associated with a level of risk acceptance authority required to meet the program risk goals.


Decreasing Probability
Decreasing Severity
Decreasing Risk

Risk Assessment Matrix Example

Probability Severity	Frequent	Probable	Occasional	Remote	Improbable
Catastrophic	1	2	3	6	10
Critical	4		7	11	12
Marginal	8	9	13	14	15
Negligible	16	17	18	19	20

Rotated Hazard Risk Index (HRI) example shown - HRI > 9 = Acceptable Risk

Using the RAM, safety risk can be managed for every hazard cause identified by assessing the severity of outcome and probability of occurrence. The corresponding risk zone determines whether risk is acceptable or additional risk reduction actions are necessary to reduce residual risk. It also defines who can approve risk which falls into the unacceptable zone.

Easy peasey, right? Not so fast.

Development programs deal with potential events and may use new parts and processes that are outside legacy experience. Here, the severity has to be evaluated at "worst case plausible outcome" and probability may need to be based on qualitative (MIL-STD-882 Table II) rather than quantitative guidance. One design engineer called this "squishy" logic, maybe due to their previous sustainment safety exposure.

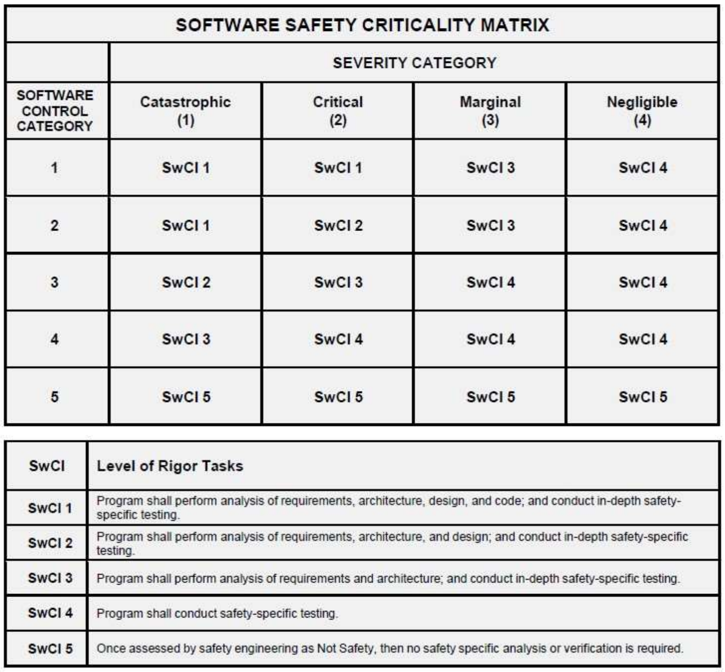
I consider it a potential red flag if a contract features severity category and RAM definitions (Tables I & III) lifted verbatim from MIL-STD-882. This and failure to gain RMA concurrence from regulators or product stakeholders can create a mud pit occluding the clear vision needed to adequately and efficiently manage product safety.

The potential outcome being knowable and/or unacceptable product risk making its way into the user community.

Chapter 5 - Software Criticality Index (SwCI)

During my time learning MIL-STD-882, one of the most vexing concepts to wrap my mind around was the level of rigor (LOR) for software testing. This is implemented as SwCI in MIL-STD-882 (Table V).

TABLE V. Software safety criticality matrix



SOFTWARE SAFETY CRITICALITY MATRIX

	SEVERITY CATEGORY			
SOFTWARE CONTROL CATEGORY	Catastrophic (1)	Critical (2)	Marginal (3)	Negligible (4)
1	SwCI 1	SwCI 1	SwCI 3	SwCI 4
2	SwCI 1	SwCI 2	SwCI 3	SwCI 4
3	SwCI 2	SwCI 3	SwCI 4	SwCI 4
4	SwCI 3	SwCI 4	SwCI 4	SwCI 4
5	SwCI 5	SwCI 5	SwCI 5	SwCI 5

SwCI	Level of Rigor Tasks
SwCI 1	Program shall perform analysis of requirements, architecture, design, and code; and conduct in-depth safety-specific testing.
SwCI 2	Program shall perform analysis of requirements, architecture, and design; and conduct in-depth safety-specific testing.
SwCI 3	Program shall perform analysis of requirements and architecture; and conduct in-depth safety-specific testing.
SwCI 4	Program shall conduct safety-specific testing.
SwCI 5	Once assessed by safety engineering as Not Safety, then no safety specific analysis or verification is required.

This was especially true when trying to address this from a system safety perspective and not as a software SME. My software background is in application development and not real-time control. It's helpful, but quite different.

The basic concept behind LOR is deceptively simple; the more critical the function, the more rigorous the testing needed before putting it into a product. The rub is that defining and implementing SwCI takes a lot of coordination between the software, safety and engineering SME's. Something not easily done as each sees the world a little differently and talk in different acronyms. The situation is not helped by terms which seem similar in concept such as DAL, CSIL, etc. but don't have accepted common mapping to SwCI.

MIL-STD-882 implements SwCI by populating a matrix very similar to the RAM (see previous post) to use severity category and something called the Software Control Category (SCC) to assign the appropriate level of testing. The SCC is intended to define hierarchical levels of software control. The example in MIL-STD-882 is just that, an example. And maybe a dated one. Tailoring should include product-specific concepts such as AI (coded vs generative), fault accommodation modes or health monitoring functions (to name a few) if used.

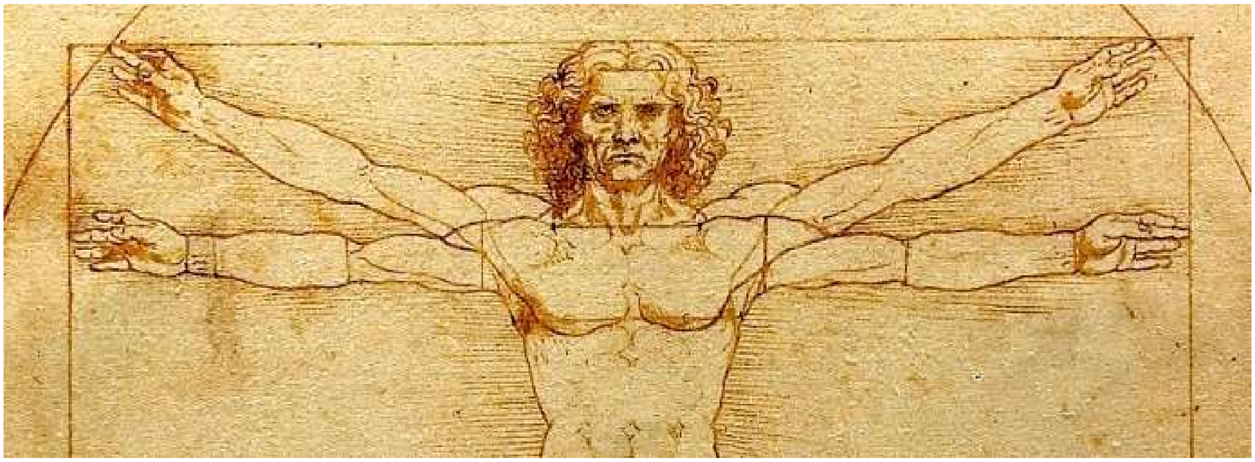
Like the RAM, the SwCI matrix is intended to be tailored to the specific product and not lifted verbatim from MIL-STD-882. Consensus is needed between the engineering, safety and software teams on what the different SCC levels are, combined with the severity of the associated software malfunction and determine what the minimum level of software testing is needed for each combination to adequately test the software prior to implementation in a product.

One solution is to treat all software as SwCI=1 and put everything through every type of software test available. But as software gets more complex and costly, testing of updates can get expensive, especially if regular software updates are needed (as with health monitoring and prognostic codes).

This creates an economic driver for a higher fidelity approach to testing rigor.

Even once this is figured out, how to apply it isn't clear. While many have looked at applying SwCI at the CSCI level, I have found the best integration with the safety process is to assign the SwCI at the requirement level. This allows the HTS to capture the assignment logic using the hazard context. However, I was unable to complete the safety/engineering/software coordination at the time of my retirement. So ironically, it's "untested".

Chapter 6 - Human Factors and Safety (O&SHA)



In the era of "do more with less", not all aerospace companies have access to Human Factors (HF) practitioners. As a result, some well-intentioned employees may venture outside of their professional swim lanes to "help out" and fill the technical gap in this niche field of science.

Many design experts tend to think that for specialized disciplines such as safety, reliability, maintainability and especially HF, knowledge is intrinsic to the design process and can be accomplished without engaging trained professionals. While some individuals may provide decent support, potential safety issues can be overlooked or actions improperly implemented. This can lead to actual practitioners being called in after problems have gotten out of hand, limiting the range of responses or cost-effective solutions. Worse, "helping out" in these disciplines can become the expectation for the technical employees that follow, eroding the discipline over time.

Along these lines, one of the more misunderstood analysis tasks of MIL-STD-882 is the O&SHA (Operating and Support Hazard Analysis - Task 206). It is a task engaged during the product development phase and includes consideration for how the operators and maintainers will interact with the system either in the creation of hazardous scenarios or how they respond to them.

In the O&SHA task, the HF discipline is front and center with many of the resulting safety actions helping to populate operator and maintenance manuals long before the product is fielded. HF consideration can begin at the concept phase.

Failure to conduct the O&SHA with full engagement of the HF practitioners can lead to significant problems downstream. As with other disciplines, HF practitioners have their own unique approaches and specialized analyses. For example, new analysis forms such as STAMP-based STPA (Systems Theoretic Process Analysis) and other structured analytical approaches exist which can expose causal scenarios for "human in the loop" vulnerabilities which could have potential safety impact.

For integrated systems, coordination with HF practitioners from participating contractors and suppliers is critical as hazard identification and corrective action determination has to ultimately be seen from the fully integrated system perspective. Fellow HF practitioners can interact with one another efficiently to assure issues are more thoroughly addressed. To do this, HF engagement with the engineering, safety, reliability and maintenance teams is a must. And the earlier this occurs in the design process, the better.

The O&SHA assures the highest level of consideration for avoiding hazardous scenarios during product use and maintenance as well as how to most effectively and safely respond should they occur.

Chapter 7 - Ground Rules and Assumptions (GR&A)

GR&A is the right path to a great safety effort

There are a number of constructs described in MIL-STD-882 that probably warrant more focused attention and description than what exists in the current version of the system safety best practice.

One of these constructs is called out in multiple task descriptions (108 and 204 through 210) which direct the documentation of assumptions made during the conduct of the safety effort, particularly during safety analysis. I found that documenting assumptions was necessary throughout the entire development process, ideally starting with the earliest concept phase.

One of the realities of product development for contractors, subcontractors and suppliers is that you typically need more information than what you have in order to do the safety work. Well written requirements leave room for interpretation (and innovation!).

But waiting for responses to questions can sometimes bring a major project to a standstill. In the case of unclear direction or the absence of answers to critical program questions, the RMA may need to make key safety assumptions (i.e., "broadcast in the blind") in order to move forward with the safety effort. The GR&A is the proper place to capture these assumptions.

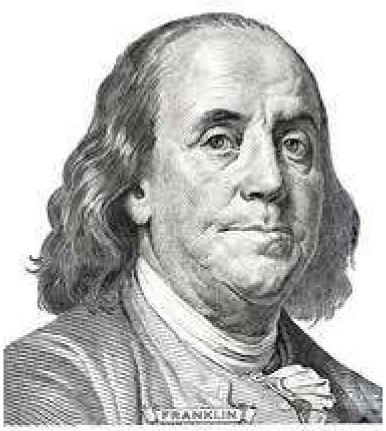
Even in the case of receiving straightforward direction, having a place to document the answers outside of typical contract documents is of great value. Further values is found as personnel on all sides of contract activity come in and out the project.

I found that the GR&A is best implemented as a formal configuration-controlled document shared amongst all stakeholders which may be updated regularly as the program evolves with product design decisions. Delivering it with all formal safety documents supporting regulator approval provides helps the safety effort downstream by providing key documentation for future reference.

In summary, a GR&A is a centralized resource which allows for timely resolution of issues that arise during the product development effort, can keep a project moving forward and provides a "point of change" should requirements evolve or safety personnel change.

That is, assuming I'm correct!

Chapter 8 - The System Safety Program Plan (SSPP)



Benjamin Franklin once said "If you fail to plan, you are planning to fail".

The authors of MIL-STD-882 may have had this in mind when they placed the SSPP (Task 102) early in the 100 series (Management) tasks section of the system safety best practice.

To me, the SSPP is the fulcrum of the system safety effort. If well executed, it's the best leverage point for an efficient and effective system safety program. It is at its peak power when it is a formal document which is concurred (preferably signed) by all key stakeholders at the start of a product development contract. It can assure that all of the "missing links" described in the previous 6 posts in this series are present.

The fundamental intent of the SSPP is to describe the system safety effort to be followed during the conduct of a product development contract.

Starting with a product definition, CONOPS, and a set of high level safety requirements, a SSPP should be unique to each product safety effort. While it is helpful to work from a previously generated SSPP for a similar product, the temptation to "force fit" a legacy SSPP into a new project in the giddy early stages of contract award can be hard to resist. While it is not uncommon for the SSPP to evolve as the program matures, starting a safety process off in even a slightly bad direction can become increasingly difficult to correct later on, especially if new analytical tasking is needed.

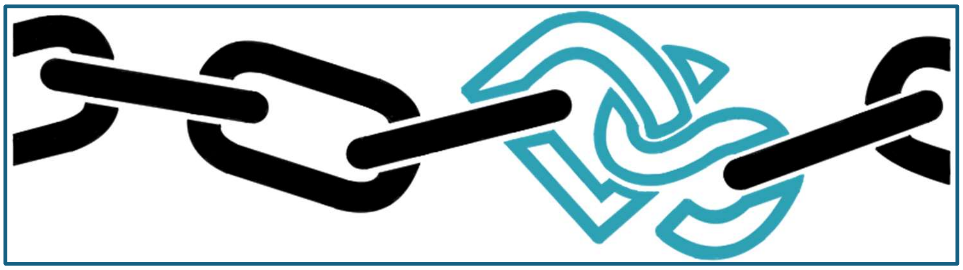
Lack of an approved and coordinated SSPP can mire a system safety effort in bureaucracy by creating needless churn and repeated meetings on things that could have been resolved from the start. This could lead to confusion and inefficiencies in the safety process or worse.

As a minimum, the content of the SSPP should include the following;
- Consideration for all applicable industry best safety practices as well as any internal (corporate) safety policies
- The system description, the major tasks to be accomplished, a description of the safety process to be followed, organizational engagement (roles and responsibilities)
- A summary of the safety metrics, project milestones and safety deliverables and a brief description of the plan for how compliance will be achieved.

Typically the RMA (see earlier post) authors the SSPP for each individual product. It would be expected that each contractor, sub-contractor and supplier would each create their own SSPP unless their contribution to safety performance is negligible. For integrated systems, the procuring RMA should have approval authority for all contractor, sub-contractor and supplier SSPPs to assure safety goals will be met.

Putting in the time up front to plan for a thorough and well-structured safety effort could save you more than a few Franklins!

Chapter 9 - Reforging the Links



It's clear to many who are looking at the industrial landscape that the system safety discipline is in crisis. Facing vexing challenges with competing safety definitions, standards, certifications, societies, approaches and experts makes it even more confounding and intimidating.

In my past two series of posts, The System Safety Challenge and The Missing Links of Safety, I've simply sought to bring my perspective to the discussion. In the comments I've received incredible perspective, some very good dialog, a deeper appreciation for what my fellow safety professionals are facing, and yes, some venting. There's always going to be an emotional component when your job carries the potential impact of saving lives.

While I don't claim to have all the answers, I do see the safety lessons learned from the post-WWII ICBM development program as a "stake in the ground" for the current system safety profession to be measured from. And from my perspective, there has been both progress and erosion in the intervening years. The initial learnings around independent oversight and safety process discipline seem to be declining. And like the system safety pioneers of the ICBM era, we are facing the challenges of new unknowns.

Maybe we need to go back to the fundamentals.

When I came into my safety position nearly 20 years ago, I opted to take a "blank sheet of paper approach" to the system safety process using my design engineering background, MIL-STD-882 (Rev C at the time) and some solid mentoring to guide my way. I used my IT background to create a software tool to facilitate the process. With that experience came no small amount of clarity for the system safety task during the development phase (my "swim lane"). That clarity extended to the role of those outside the safety team, all the way up to the executive level. Also, no small amount of empathy and compassion for my fellow safety professionals.

My mission going forward is to share that knowledge to help the industry in any way I can. Certainly helping to teach system safety at USC is both a passion and a big part of that mission. To add to that, I am happy to announce the creation of DevSafe Consulting, LLC to provide a platform to allow a more focused application of the lessons I've learned from my 40+ years in the aerospace industry.

I know there are other voices that need to be heard, but we need to figure out how to work together to face the many challenges. None of us should compete on safety. As a good coach implores his team, we need to work the fundamentals. If we can agree on what those fundamentals are, we can grow from there.