Timing of Formal Phase Safety Reviews for Large-Scale Integrated Hazard Analysis

Michael J. Massie[*]
ARES Corporation, Houston, Texas 77058

A. Terry Morris[†]
NASA Langley Research Center, Hampton, Virginia 23681

Integrated hazard analysis (IHA) is a process used to identify and control unacceptable risk. As such, it does not occur in a vacuum. IHA approaches must be tailored to fit the system being analyzed. Physical, resource, organizational and temporal constraints on large-scale integrated systems impose additional direct or derived requirements on the IHA. The timing and interaction between engineering and safety organizations can provide either benefits or hindrances to the overall end product. The traditional approach for formal phase safety review timing and content, which generally works well for small- to moderate-scale systems, does not work well for very large-scale integrated systems. This paper proposes a modified approach to timing and content of formal phase safety reviews for IHA. Details of the tailoring process for IHA will describe how to avoid temporary disconnects in major milestone reviews and how to maintain a cohesive end-to-end integration story particularly for systems where the integrator inherently has little to no insight into lower level systems. The proposal has the advantage of allowing the hazard analysis development process to occur as technical data normally matures.

Nomenclature

CDR	=	Critical Design Review
CEV/CLV	=	Crew Exploration Vehicle / Crew Launch Vehicle
DOD	=	Department of Defense
FAA	=	Federal Aviation Administration
IHA	=	Integrated Hazard Analysis
ISS	=	International Space Station
KSC	=	Kennedy Space Center
NASA	=	National Aeronautics and Space Administration
NSTS/STS	=	National Space Transportation System / Space Transportation System
PDR	=	Preliminary Design Review
PSRP	=	Payload Safety Review Panel
SDR	=	System Definition Review
SE&I/SAVIO	=	Systems Engineering & Integration / Software and Avionics Integration Office
SR&QA	=	Safety, Reliability and Quality Assurance
SRR	=	System Requirements Review

I. Introduction

Traditional IHA phase safety reviews work well for small- to moderate-scaled systems. The inherent structure, unfortunately, breaks down when dealing with large-scale integrated systems. This paper will propose a solution to this problem by detailing both how to tailor the timing and how to modify the content of the formal phase safety review in order to provide technical coherency and viability to the program including the organizations providing the engineering, the integration as well as the safety of the overall end product.

[*] Lead for NASA's Constellation Integrated Hazard Analysis, ARES Corporation, Johnson Space Center

[†] Safety-Critical Avionics Systems Branch, Mail Stop 130, SAVIO IHA Lead, AIAA Lifetime Associate Fellow.

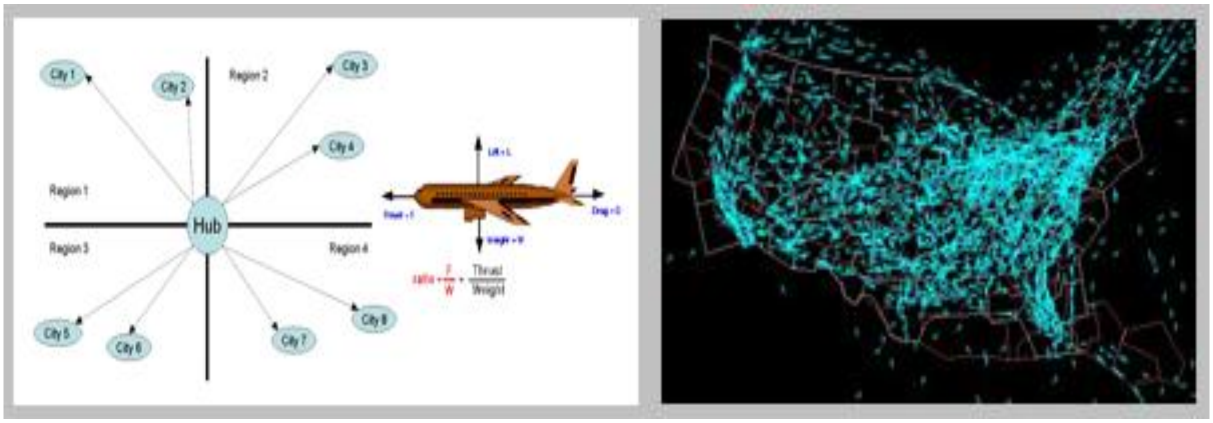


Figure 1. FAA System of Systems Example[1]

II. Background

A. Very Large-Scale Integrated Systems

Large systems of systems type programs have been around for decades as various federal departments and agencies of the United States (FAA, DOD, NASA and others) adapt to our changing world. Modern fighter planes, modern commercial jetliners, nuclear submarines and aircraft carriers are all large complex integrated systems of systems where multiple dedicated systems pool their resources and capabilities together to obtain a new, more complex system which offers more functionality and performance than simply the sum of the individual systems. Figure 1 displays one example of a system of systems for the FAA. These systems must coordinate their efforts and must work together to carry out a mission or achieve a national objective. Nuclear submarines and aircraft carriers, for instance, must support and sustain a small city of people while executing their various missions. Life support systems, power, thermal control, and propulsions systems must be combined to create a vehicle that can support the people and the associated processes to accomplish various goals. As the DOD and FAA responsibilities continue to grow, for example, it becomes very necessary to network existing assets together and to manage/control these assets more securely. This increased responsibility coupled with advances in technology led the push toward rudimentary large scale system of system programs. There was clear understanding in the initial formation of these programs that communication was the key to providing knowledge of location, health & status, caution & warning, battle readiness and other information in order to allow a coordinated effort between many heterogeneous components. The FAA, for instance, had to control more and more aircraft takeoffs and landings using the current hub-spoke system (see figure 1 left) and had to simultaneously monitor aircraft altitude and positions across the country in order to reduce the risk of aircraft collisions. One of the metrics the FAA uses to quantify aircraft safety is aircraft spacing. As the number of airline flights has increased over the years, the number of simultaneous aircraft in the national airspace has increased (see figure 1 right). This presents a complex and daunting task for the FAA since increasing the number of simultaneous aircraft in the airspace decreases the margins allotted for aircraft safety. As similar large scale type systems continue to grow in use and to expand in capability, the interactions between the various subsystems grow more and more complex. In many of these cases, the interactions not only behave differently (as compared to small- or medium-sized systems), they actually conflict giving rise to the need for improved management strategies for large scale systems.

Very large scale integrated systems are created when numerous dissimilar assets are linked together not only via communication interfaces but also by physical and functional interfaces. To add to the complexity, some of these fundamental interfaces are purposely or inadvertently connected and disconnected during a program's life cycle. This introduces a very important dimension when developing large scale integrated systems. This dimension is timing. Determining when each subsystem interfaces and influences other subsystems is just as important as determining how each subsystem interfaces with the others.

Let's use the Pyramids of Giza as an example, even though they are very large structures which took decades to complete, they are not really considered very large scale integrated systems by today's standards. The fundamental building block the pyramid architects used was a basic stone block. Though each stone block was tailored to do its job, the architects designed the block pieces to integrate together and to give rise to the singular pyramid structure.

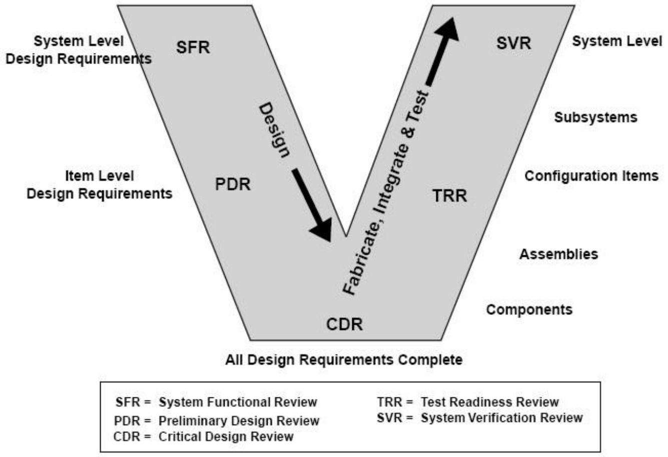

System Level
Design Requirements
SFR
Design
Item Level
Design Requirements
PDR
CDR
All Design Requirements Complete
Fabricate, Integrate & Test
SVR
System Level
Subsystems
Configuration Items
TRR
Assemblies
Components
SFR = System Functional Review
PDR = Preliminary Design Review
CDR = Critical Design Review
TRR = Test Readiness Review
SVR = System Verification Review

Figure 2. Generic System Engineering Life Cycle “Vee” Model[3]

Assembly was ordered and, for the most part, synchronized between stone block production and readiness to accept and place the blocks at the worksite. The timing of the blocks for assembly and installation had a significant bearing on the amount of resources utilized in pyramid production. Timing is even more critical when considering today’s large scale systems. While the building of a battleship may appear somewhat asynchronous, all of the parts must be made available in a particular order so that the ship manufacturer can ultimately assemble various copies of the same ship in an efficient manner. Very large scale systems are made up of numerous blocks that are dissimilar and in fact are each unique to do a specific job where each piece is contributing its function to achieve overall program objectives.

One-of-a-kind builds for large scale integrated systems create very diverse hardware development processes. This is because of the unique nature of the task. For small to moderate scale systems, like televisions or cars, the industrial build process can be developed as an assembly line process where the architect designates when and where each component will be integrated. For many large-scale, one-of-a-kind systems, it is extremely difficult to get the parts to arrive in a given order since the system has never been developed before and the replication process has not been conceived nor optimized. To compensate, program managers usually adapt system integration strategies to the ability of each component being delivered. Programs with these features generally employ lots of simulators to not only test the capabilities of one major unit but also to allow for testing of a unit while other units are still in development. Two examples of these one-of-a-kind systems include the International Space Station (ISS), tasked to support human endeavors in low earth orbit, and NASA’s Constellation program, tasked to return humans back to the moon. ISS is the first example of a very large integrated system where 44 one-of-a-kind elements were combined into one unique vehicle that could support up to seven crew and accommodate up to seven more visiting crew while providing all of the capabilities needed to run experiments and sustain human lives. NASA’s Constellation program is another very large integrated system which is developing launch vehicles and space transportation vehicles unique to their coordinated missions to carry the crew to the ISS, the moon or even Mars[2]. According to the Constellation program plan, numerous unique physical and functional interfaces will be developed at different times for different phases of the overall mission. Numerous projects or systems, for instance, will be initiated at times when their unique contracting and setup activities can get underway. Thus, in many cases, the sequencing of the project milestone reviews cannot be performed in step with the program milestones.

Large scale integrated systems also suffer from the lack of clear terminology specifically with respect to what is called a system, a program , a project, an element, etc. For the purposes of this paper, two sets of terms will be used: one for ISS and the other for Constellation. NASA generally uses the term ‘program’ to refer to the highest level of a system. The individuals who manage the overall system are called the ‘program office.’ Both ISS and the Constellation programs have separate and distinct program offices. NASA allows each program to decide how to partition and name the various components of the program. In the case of ISS, the major components are called ‘elements.’ In the case of Constellation, they are called ‘projects’ or ‘systems.’ Constellation managers have decided to call major components of each project an ‘element.’

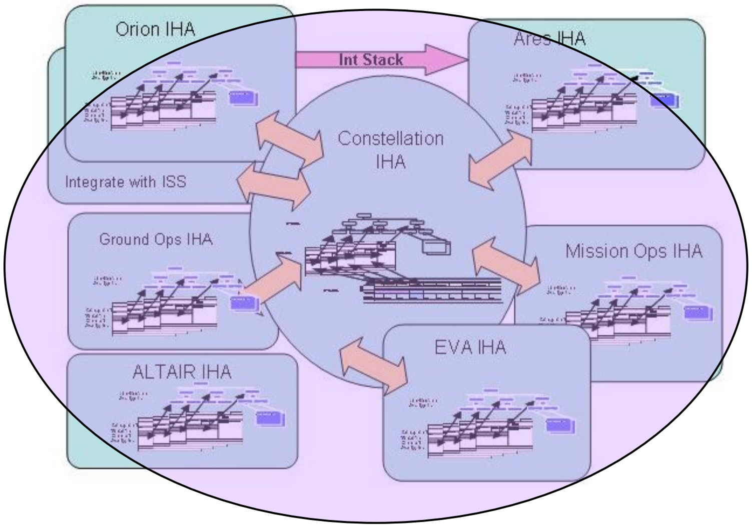


Figure 3. Constellation's Integrated Hazard Analysis Structure

B. The Integrated Hazard Analysis Process

In order to manage the complexities of large scale integrated systems, many organizations develop variations of the system engineering life cycle model (see generic example in figure 2). Generic system engineering processes follow the divide-conquer-integrate paradigm. That is, they attempt to formally analyze the large scale system by partitioning into smaller sequential steps like design, fabricate, integrate and test. Major decision points are integrated into the analysis to provide program managers periodic insights into the risks associated with the overall system. These decision points are called program reviews. Each program outlines the types of reviews it expects depending on the type of integrated system being developed. NASA, for instance, utilizes its NASA procedural requirements 7120.5D and levies them on all programs[4]. The Constellation program, for example, abides by this procedural requirement and utilizes the milestone reviews identified for human space flight projects. The reviews that are applicable to Constellation are system requirements review (SRR), system definition review (SDR), preliminary design review (PDR), critical design review (CDR) and flight readiness review to name a few.

In addition to program reviews, each organization developing a large scale system also establishes a system safety process to ensure that risks are managed throughout the particular life cycle. To be effective, the system safety process should employ an iterative hazard analysis process so that all available data are analyzed early and often for potential hazards. Only in this way can safety be designed into the product and insights into potential undesirable behavior be revealed early enough to prevent or to mitigate the hazardous behavior by way of effective redesign. The hazard analysis process specifically iterates between the hazard causes and the hazard control story in such a way as to realize a system with acceptable risk. Massie, IHA lead for NASA's Constellation program, has prescribed four keys to success that if adopted will lead a system safety analyst to accomplish the enormous task of integrating various systems iteratively in a large scale distributed system[5]. These keys reveal strategic, operational and organizational lessons learned from previous IHA experiences including NASA's ISS program. The four keys are: 1) define the analysis structure, 2) provide a good IHA plan, 3) provide for good and reliable communications and 4) select and utilize the right personnel for the job. These steps were applied to NASA's Constellation program where the Orion crew exploration vehicle (CEV), the Ares I crew launch vehicle (CLV), mission operations, the astronaut suits and ground operations were integrated across a mission timeline with the purpose to dock the Orion space vehicle with the Space Station and safely return the crew to earth. The structure of Constellation's IHA involves analyzing the undesirable interactions between the projects that could lead to a hazard (see figure 3). The IHA process involved identifying the integrated system-level risks. This involves identifying all hazards from a systems perspective, identifying all causes to each hazard, revealing the control story and providing verification of

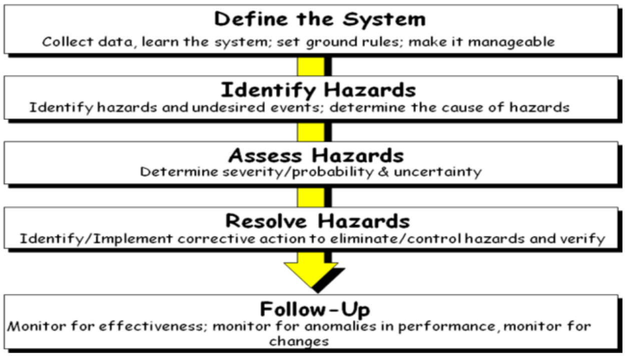

Define the System
Collect data, learn the system; set ground rules; make it manageable
Identify Hazards
Identify hazards and undesired events; determine the cause of hazards
Assess Hazards
Determine severity/probability & uncertainty
Resolve Hazards
Identify/Implement corrective action to eliminate/control hazards and verify
Follow-Up
Monitor for effectiveness; monitor for anomalies in performance, monitor for changes

Figure 4. Systematic Hazard Analysis Process

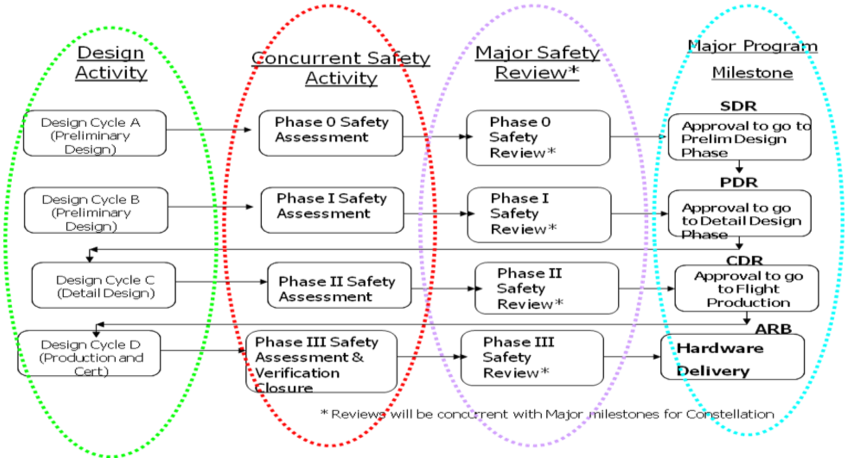

Design Activity
Concurrent Safety Activity
Major Safety Review*
Major Program Milestone
Design Cycle A (Preliminary Design)
Phase 0 Safety Assessment
Phase 0 Safety Review*
SDR
Approval to go to Prelim Design Phase
PDR
Design Cycle B (Preliminary Design)
Phase I Safety Assessment
Phase I Safety Review*
Approval to go to Detail Design Phase
CDR
Design Cycle C (Detail Design)
Phase II Safety Assessment
Phase II Safety Review*
Approval to go to Flight Production
ARB
Design Cycle D (Production and Cert)
Phase III Safety Assessment & Verification Closure
Phase III Safety Review*
Hardware Delivery
* Reviews will be concurrent with Major milestones for Constellation

Figure 5. Overview of Constellation Phased Safety Review Process

the controls in a systematic and iterative fashion (see generic flow in figure 4). According to Constellation guidelines, hazard-related risks are to be identified during phased safety reviews which occur around major program milestones. Hazards, for instance, are identified during the program's initial design cycle, causal contributors are identified during the system definition review, controls are identified and analyzed during the preliminary design review with the verifications outlined during the critical design review (see figure 5). At each major review, the goal of the iterative hazard analysis is to ensure that the system stays in a known safe state regardless of the mission timeline or the state transitions. The hazard analysis also attempts to update system requirements or system-level functions to ensure that the system remains in a known safe state according to the program's level of acceptable risk.

C. Significant Roles within the IHA Process

Because of the size of large scale systems, organizations typically divide organizational responsibilities as well as develop processes so that each organizational function can contribute to the end product. From an IHA perspective, there are three primary roles that interact to reduce hazards in large scale systems. These are the engineering role, the safety role and the integration role.

1. *The Engineering Role*

Each program will divide its major design function by the typical engineering disciplines. At the program level these engineers become the leadership for the design of the integration of the hardware/software and operations for their discipline. Their roles may be delegated or shared with other developers/projects but they become the de-facto experts for the data and program progress in their discipline.

2. The Safety Role

The safety analysts become the authors of the hazard reports and partner with the leader of the engineering design discipline to create an accurate hazard analysis of the design. They will work with each of the discipline efforts to assure that the design meets the intent of the applicable safety requirements. They also help the design team to document their results and aid the team by presenting those results to the safety review panel. The success of the hazard analysis is totally dependent on the interactions of the hazard report authors and the design owners. A second aspect of the safety role resides in the safety panel, the group of experts who review the author's hazard reports and provide constructive feedback to the authors to ensure that the analysis is conducted properly and to critique the risk results.

3. The Integrator Role

The hazard analysis integrator is the chief architect behind the hazard analysis and helps the hazard analysis team develop an architecture that best highlights the risks of the program, and assures that coordination of hazard analyses that overlap or touch on similar subject areas are coordinated. The integrator assures that functions are assessed for hazards end-to-end without regard to who owns particular hardware/software components with a perspective that incorporates both sides of the interfaces. Additionally, the integrator assists the hazard analysis authors by guiding/directing/instructing them how to deal with the asynchronous development of the lower level data that is required to support the IHA.

D. The Necessity of Healthy Interaction between Engineering, Safety and Integration

It is healthy to exhibit a moderate amount of friction between engineering and safety personnel, particularly when the end product is high risk. The engineering team wants to meet performance objectives while the safety team tries to ensure that hazards do not occur. This relationship between the engineering design and safety personnel must be fostered into a push-pull relationship where the designer is looking at the design from the standpoint of why it works and the hazard analyst is looking from the why it might not work standpoint. As they work together they must optimize the design to work as intended while simultaneously putting as much margin in as possible to preclude inadvertent functions from working when they are not needed and to prevent intended functions from working in unintended ways that could cause harm to the vehicle, the system or the crew.

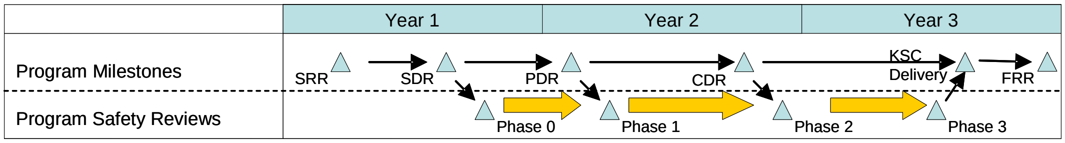
E. Timing and Content of the Traditional Phased Safety Review Process

Overview of the phased safety review process has already been depicted in figure 5. The interactions of the IHA projects used on NASA's Constellation program has already been shown in figure 3. The program interface (in the middle of figure 3) represents the primary integrator, that is, the organization responsible for integrating the lower level control story so the that program achieves a coherent, integrated end-to-end hazard story with respect to hazards.

The genesis of the phased safety review process comes from the National Space Transportation System (NSTS) payload safety review process documented in NSTS 13830, Payload Safety Review Process[6]. This process was developed to allow NASA's Shuttle Program to review and evaluate the safety of experiments to be flown on Space Shuttle Missions. The process was intended to iterate with the payload design process and require the payload or experiment developer to bring to the Payload Safety Review Panel (PSRP) the information necessary to allow the PSRP to provide timely input to the developer so that they could design in adequate safety features that preclude the experiment from causing harm to the Shuttle or the crew. The Payload Program and Safety Review Process flow is shown in figure 6.

Each phase of the payload safety review process brought the panel and designers together to discuss the needed safety features commensurate with the maturity of the program. Figure 2 provides a general description of the typical development flow of a program. Program milestones are associated with each program development step where each step has an associated phase safety review set of expectations.

System definition review (SDR) is a requirements baselining process and so the phase 0 safety review was used as a technical interchange meeting to discuss the incorporation of the payload safety requirement into the program requirements. This meeting was also used to familiarize the PSRP with the design and operations concept and to



Benefits

- Safety Evaluation of Program based on Milestone baseline Configurations
- Design & Safety Engineers Efforts Support Program Milestones then Safety Reviews
- Process is Orderly and Iterative

Drawbacks

- Safety Reviews after Milestones
- Designers must support re -review design just approved at milestone
- Risk of Safety Review Resulting in Changes to milestone baseline
- Perception Safety Input too Late to make cost effective changes

Figure 6. Typical Payload Safety Review Process

help the designers understand the intent of each of the safety requirements. Typical phase 0 safety reviews require the hardware providers to demonstrate that they understand the safety processes and the safety requirements and that they understand safety analysis sufficiently to identify their preliminary hazards and causes.

Preliminary design review (PDR) is a process that validates that the preliminary design closes around the requirements. The phase 1 safety review was developed then to evaluate the design concepts ability to close around the safety requirements. Criteria for the phase 1 safety review is to have the hardware developers demonstrate that they have evaluated the design such that the hazard control approach will adequately satisfy the applicable safety requirements for each of the previously (phase 0) identified hazard causes. Any gaps in the analysis or weakness in the control approach were addressed as needed areas for design concept change. Phase 1 criteria also required the hardware developer to peek ahead at the plans for verification of the hazard controls and to identify the methodology intended to be used.

Critical design review (CDR) is the acceptance of the detailed design and the beginning of design verification. At this phase 2 stage, the actual hazard controls are to be identified and validated to be a real part of the design. In addition, the hardware developer has to show the detailed verification plans for each hazard control and assure that those plans are comprehensive and adequate to assure that the controls are real and operate as intended in all expected environments and operational situations.

Hardware delivery to Kennedy Space Center (KSC) launch site typically signified the completion of all design development manufacturing and test processes for the payload or experiment. Thus prior to delivery, the safety review process required a final review (phase 3) meeting with the hardware developer to assure that the hardware/software and operations verifications went as planned and to deal with any anomalies in the verification program results. Any hazard controls verifications that could not be accomplished before delivery (eg: payload to Orbiter separation system safing for deployable payloads) were tracked on a verification tracking log with results reported back to the PSRP and the program prior to flight.

III. Tailoring the Timing and Content of Formal Phase Safety Reviews

This section will describe the challenges in applying traditional phase safety reviews on two large scale integrated systems, will propose a modified half cycle safety review process and will discuss the risks associated with such a tailoring.

A. Challenges in applying the Traditional Three Phase Safety Reviews to a Very Large Scale Program

1.0 International Space Station

Historically, NASA develops a very large scale program about once every 20 years. Because of this, it must often update its processes to the latest industry standards when developing new large scale programs. Apollo (circa 1960's) was the 1st large scale NASA program, followed by the Space Shuttle program. These programs incorporated safety via the application of national and local standards for hardware development relying heavily on standards from the American National Standards Institute, Underwriters Laboratories, the National Institute for Occupational Safety and Health and many other national organizations of the time. The design analysis processes included failure modes and effects analyses and some early forms of hazards analysis but relied heavily on fault oriented techniques and processes of the day. As Shuttle was completing development, MIL-STD-882[7] was becoming the military standard for safety review processes and NASA developed the Payload Safety Review Process described in the prior section. However, treatment of very large scale program safety for space was still a developing concept as the International Space Station was being conceived and initiated.

The International Space Station wasn't just a large object like one of the Pyramids of Giza being launched into Orbit. It is truly as large integrated system of systems with dozens of developers both large and small. The job of the integrator was to get this program (with all of its independently designed and developed parts) to work together in such a way as to provide a safe, long lasting and reliable habitable environment in space. The Shuttle program already developed the payload safety process so this process was adopted to facilitate the safety of the emerging International Space Station design. However, it eventually became apparent that there were four major obstacles that had to be analyzed, understood and tailored before simply applying the traditional payload safety process to ISS development.

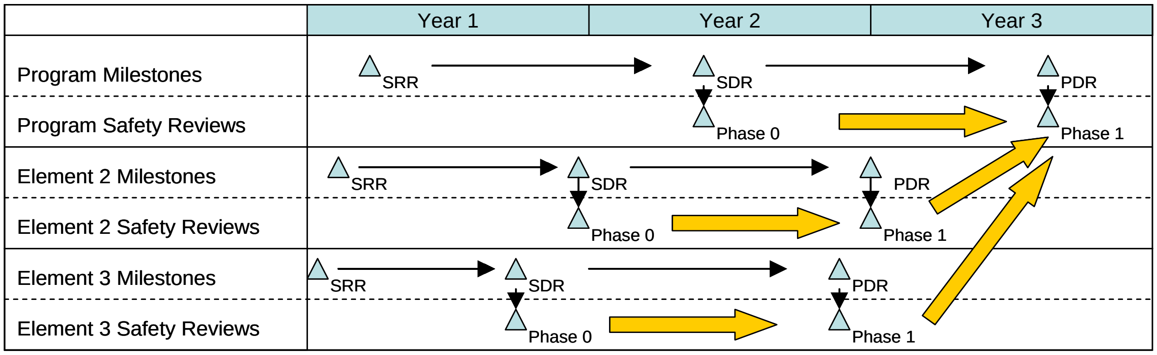
First and foremost, ISS wasn't just a program but a program of programs. Each module was a major program in it's own right that required full scale development engineering processes. With a planned 44 missions to develop, build and expand ISS, there needed to be 44 sets of three phased safety reviews. Very large scale programs accomplish what they can accomplish when they can accomplish it. However, they usually undergo many schedule iterations and even complete revisions as they get the different development and manufacturing operations underway. As the design progresses a common architecture and design arises. This design emerges through systematic top-down development efforts and via iterative development cycles within the program architecture. This concurrent and iterative development approach created huge challenges for the traditional payload safety review process. 132 Safety reviews over the initially planned development period of about 10 years was barely achievable with typical full element reviews running up to 3 weeks in length. The simultaneous start of most elements put huge pressures on the safety review process to hold many of these reviews at the same time. Neither the panel nor the integrators could support such a schedule and get any other work done – like the actual integrated hazard analysis. Program modifications like addition of an entire new module infused still more demands on the safety review process. To further compound the problem, purely logistic flights to ISS were added to the manifest, which brought hundreds of new pieces of small hardware to the ISS but added no new functionality to the vehicle, but still needed evaluated against the safety requirements.

A second challenge for the ISS program safety review process was that the Payload Safety Review Process required a phase 3 safety review which included evaluation of the hardware/software/operations verifications prior to delivery of the hardware to the Kennedy Space Center for launch processing. However all of the hardware was being delivered to the Kennedy Space Center with the intent of complete assembly and test operations to be performed. Later the program added integrated testing between the elements which was to be accomplished by functionally attaching as many elements/modules together as could be accommodated and running or enabling the critical systems. This left the safety review process with no hard review point at which to review the testing and verification results before these modules would be flown. In fact, some final verification could not be completed until the modules were attached and checked out on orbit.

A third challenge to the payload safety review process for ISS was the sheer volume of paper involved. Complete traditional safety data packages for a payload might span 2500 pages for an experiment that fit in a 2' x 2' locker. The safety review process had to learn to manage data and information much better to preclude having to review a large number of hazard reports. For example, there might be 45 (counting integration) hazard reports, each reiterated the following typical comment from an ISS hazard report author, "my structure is designed to the appropriate Space Station Structural Design Standards and meets the 1.5 Factor of Safety with a positive Margin of Safety."

Fourth, the ISS program did not go through the traditional SDR, PDR and CDR development cycles. ISS was based on a more integrated incremental design review process and then a series of system integration reviews related to the operations for each flight. With so many elements in various stages of development at different periods of time there wasn't a true PDR/CDR at the program level to address the integrated design in a traditional phased safety review manner.

Thus, in the case of the ISS program, asynchronous hardware development timing was a major challenge to holding the traditional phased safety review processing for the integrated system. This challenge manifests itself differently in every large scale program and thus requires unique solutions to each program but this feature can help



Benefits

- Moves Safety Evaluation of Design to be Concurrent with Program Evaluation of Design
- Reduces Perception of Safety Coming in After the Fact

Drawbacks

- Requires Simultaneous Design Evolution and Safety Review Preparation and Documentation
- Conflicts Safety Review and Program Milestone Resources
- Does not accomplish goal of getting Safety Decisions in ahead of design

Figure 7: Concurrent Program and Phased Safety Review Milestones through Phase I – Synchronous Milestones

drive a more consistent solution between various programs if the particular issues are well considered. Large scale programs share this asynchronous approach to major element development and so the safety review process must be adapted to accommodate it.

2.0 Constellation Program

The next major large scale program NASA has pursued is the Constellation program. Constellation shares several features of very large scale programs with the ISS Program. In particular, the asynchronous development processes for each of the major elements (called Projects on Constellation). Thus, major Projects like the Orion CEV and the rocket that launches it, the ARES CLV, are proceeding on their own program development cycles independent of each other and the other major projects on the program.

CxP program has elected to proceed at the integrated program level on a more typical program development process including a program SSR, SDR, PDR, and CDR. However this cycle is based more on high level integration objectives and so is not sequenced with the project milestones of similar type. So the phase 1 safety review associated with Orion development was not completed at the time of the program level PDR and associated phase 1 reviews even though ARES had completed its hardware PDR. In fact, most of the other projects had not even started their PDRs yet. This asynchronous timing of the program/project milestones not only makes it difficult if not impossible to meet the actual traditional phase safety review criteria it also puts pressure on the hazard analysis organizations to support a process that the data does not support particularly if the process review criteria is applied rigidly.

Constellation program managers also decided that in order to create safety inputs earlier into the design they would require the Phased Safety Review Process milestones to occur in exact sync with the program milestones. See figure 7 for how a synchronized flow of this type would have to occur. Of course, with asychronized starts of the different projects this plan is not realizable. More importantly, via inspection of this simple flow one can quickly see that the more elements that are added to this flow the more compressed the support schedule for this approach becomes until enough elements are added that even a controlled synchronization schedule cannot be sustained because of too many elements trying to hold their project milestones in the same block of time. Eventually so many elements/projects are added that the system becomes asynchronous. Very large scale programs must all deal with these phenomena. In fact, asynchronous development actually allows the developers to arrive at the goal more closely together ultimately contributing to a much early design completion if managed well.

3.0 The factors that drive an acceptable solution

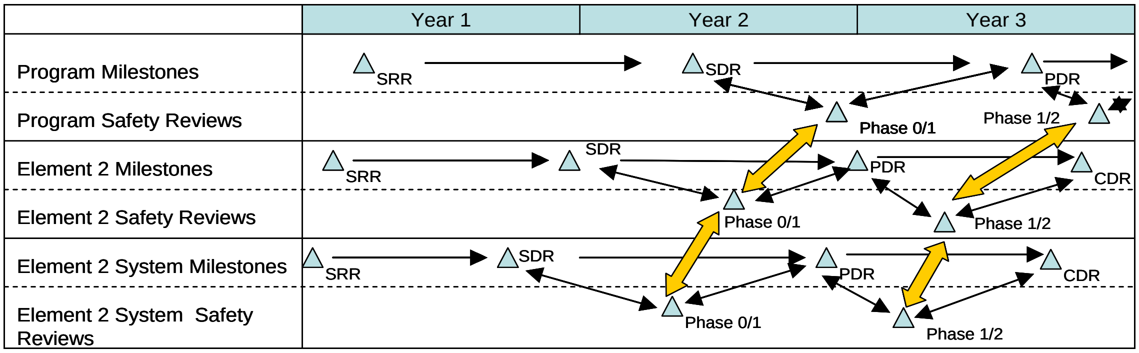
The hazard analysis processes itself must run in parallel with the program (see figure 5) so that it can take in information as its being developed, interpret it, assess its impacts and respond to the program in a way that affects the program in a useful way. Infusing ill-timed changes into the program in the name of safety has in the past brought about the very incidents that were trying to be precluded. For example, it is very difficult for the program to deal with a new hazard that has been discovered and requires mitigation shortly before final delivery of the product to the customer. The program processes for designing, manufacturing and testing the influences of new design features are not well equipped to respond to late change needs. Programs like Tethered Space Satellite[8] have suffered when it tried to implement late imposed safety features that ultimately caused major system failure because all of the impacts of the new design could not be understood before manufacturing and assembly and delivery. In this case, a late breaking change in the name of safety resulted in loss. During the first Tethered Space Satellite mission which occurred on the Space Shuttle (STS-46) in July of 1992, a protruding bolt had prevented full release of the tether. This was due to an improper late-stage modification of the deployment reel system. A re-flight of the tether system happened on February 25, 1996, where five hours after deployment, the tether cable suddenly snapped near the top of the deployment boom due to debris. The typical response to these events is for follow on programs to pressure the system safety process to provide safety input earlier. This is a logical request but must be looked at carefully to assure input is not so early that it is either of no value (due to lack of detail) or timed on top of the major program milestones such that it cannot be incorporated into the design milestones.

Ideally, the program architects need to make sure that the safety process can be fed appropriately mature data, with enough time to synthesize it and then to provide adequate input based on this data to the program describing both the expectations and safety feature needed in time for the design to incorporate them into the next major milestone. The architect of the program safety review process must then consider the need for early hazard control inputs to the program development process and how to allow flexibility in the safety review process to accommodate the asynchronous lower level project or element milestone flows. As previously explained, some programs put the safety review milestones well after the program milestones of equivalent maturity so that the safety process can be based on the program milestone mature data (see figure 6). For example, the most common approach is to hold a phase 1 safety review up to 90 days after a program PDR so that the safety process can be a full assessment of the program PDR data. This has the advantage of making the safety process input mature but provides very little input to the program on the needed changes to the design before the program milestone itself. And this approach requires delay of the program phase 1 safety review until all projects have completed their PDRs and associated phase 1 safety reviews.

Other programs accelerate the entire expectations of the safety review process, forcing the safety analysts to seek data that exceeds the maturity of the program as whole. An example of this condition would be when phase 0 expectations resemble phase 1 criteria and phase 1 safety review criteria more closely resembles phase 2 safety review criteria. This approach has the advantage of earlier notification of issues and needed design changes by advancing the safety review process ahead of the program, but most programs struggle implementing this approach because of the mismatch in data maturity and the process expectations. When combined with subsequent, asynchronous development cycles of the lower level elements/projects, successful completion of these high expectation reviews is neither realistic nor achievable.

Other approaches have also been attempted like holding the phase safety reviews concurrent with the program milestones (see figure 7), or decoupling the safety review process from the typical program milestones - all of these approaches create mismatches either in data maturity, in personnel available to support the safety review process, or in timeliness of safety inputs to the program.

The mismatch of personnel resources to support the safety review process is an often overlooked major driver to the success of the program and safety integration. This mismatch can be created by timing safety process demands and program milestone demands such that both reach peak demand at the same time and reach minimum demand at the same time. As shown in figure 7, at the program level the design organizations must review their requirements, create the design to meet those requirements, study the implications of their design and prepare reports, presentations and supporting data for the program milestone reviews. The safety analysis process requires the same thing of the safety analysts, but the two processes are codependent. Forcing them to operate in an exact step –for-step fashion creates huge demands on the design and safety analysts' time that are in perfect conflict. To support the program PDR the designers need the safety analysts to review all of the data that provides input to the design solutions (eg; lower level procurement specifications, lower level design details, interface control requirements, etc) and provide safety inputs that will affect the design and assure compliance with the safety requirements. However,



Benefits

- Moves Safety Evaluation of Design to be Before Program Evaluation of Design
- Reduces Perception of Safety Coming in After the Fact
- De-conflicts Safety Review and Program Resources

Drawbacks

- Requires Simultaneous Design & Safety Resources to assess partial design on top of design org prep work for program milestones
- Forces eval of design not yet baselined
- Safety Review & Decisions ahead of design Milestone Reviews but they are too close to milestone to affect it

Figure 8: Half Cycle Phased Safety Review Process – Synchronous Flow

at the same time the safety engineers need the designers to review their analyses and provide design details to make the safety analysis complete. But the designers are trying to create the design and the safety analysts are trying to analyze the emerging design at the same time. This creates a perfect conflict of resources and goals. Everyone is consumed with their own processes at the same time and each cannot help the other effectively.

Therefore in order to architect an effective program the design analysis and safety analysis cycles need to be synchronized to become more complementary such that one can feed the other. In addition, the asynchronous nature of very large scale program development must be accommodated for any review process to be successful. In order to solve the first problem a natural proverbial "chicken and egg" problem must be addressed. If the hazard analysis needs design information to do an assessment and the design organization needs safety input to even create an effective design it would seem both should "go first." Thus, further review of the features of both processes is required to arrive at successfully integrated program architecture. And in fact, a solution to the first problem presents an opportunity to solve the second problem.

B. Proposal: A Modified Formal Safety Review Process (in terms of Timing and Content)

1.0 Creating a Half Cycle Safety Review Process

Looking at the problem from the point of view of program-level needs brings potential solutions to light. The program needs safety criteria or input for each major milestone. The program then needs an evaluation of its ability to satisfy those requirements prior to the next milestone and needs the evaluation results in time to react to the identified shortcomings in satisfying the safety requirement before the next milestone. The safety process is setup to create evaluations of available data and provide input to the design organization for their next major design cycle. All previously discussed safety process timing solutions give rise to the mismatches in safety issue identification, program resources or safety review process expectations so a new solution could be more effective.

One way to address the mismatches is to shift the safety review process timing such that it looks backward to what's been done at the prior program milestone AND forward to what's needed for the next milestone. To look simultaneously forward and backward at the program milestones, the safety process then would have to be placed strategically BETWEEN the program milestones instead of trying to run concurrent with them. This strategy also immediately solves the resourcing conflicts because each process reaches its peak resource demands at one half cycles apart (see figure 8). This approach frees the safety analysts to participate in the review item disposition creation and resolution processes that feed the program milestones reviews and thus support the entire program

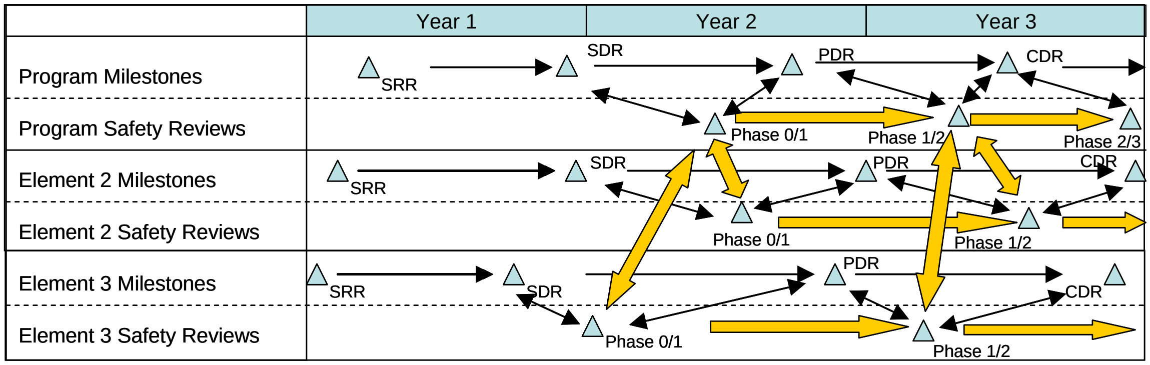


Figure 9: Half Cycle Safety Review Process – Asynchronous Flow

milestone. In addition, it calls upon the program designers to support the hazard analysis development process and the safety reviews when they are not encumbered by the program milestone review demands.

This new approach solves two of the mismatch problems very readily. So we need to investigate the potential for expectation mismatches with program data maturity. Placing the program safety review processes at program milestone one half steps requires redefining the phase safety review criteria somewhat. In fact, it requires a meshing of the prior program milestone safety criteria with the upcoming milestone criteria.

If we assume this proposal was implemented on a large scale integrated system like Constellation, the phase 0 review would become a phase 0/1 review held between SDR and PDR. The assessments that support this meeting would be based on the program SDR data and identify expectations for the program to meet for the PDR. Doing this assessment halfway between the milestones means that the program will be well on it’s way towards creating the preliminary design but not finished yet, therefore design change requests from the safety process can be readily implemented before the program PDR. Another major benefit is that the hazard analysts can then use their phase 0/1 review results to support their evaluation of the program PDR data and have a solid basis for the review item dispositions they write. This whole cycle iterates again between PDR and CDR and again between CDR and hardware delivery.

2.0 Dealing with Asynchronous Lower Level development cycles

Establishing half cycle safety reviews solves the problems of mismatched requirements and data flows between the program level hazard analysis and design organizations. In addition, this process resolves resource demand mismatches as well. And most importantly it provides an avenue for aligning safety expectations with program maturity (see figure 9).

Additionally, the process has the ability to align expectations with the hardware/software/operations maturity that allows the half cycle safety review process to address asynchronous lower level development. Because the expectations can now be defined across the spectrum of phase 0/1 expectations and later at phase 1/2, this flexibility allows the analysis to mature in synch with the program and the safety review panel to adjust their expectations for each piece of the developments maturity. Thus, it is important to establish the exact criteria for each hazard cause based on the maturity of the design of the associated hardware/software and operations. This will allow both timely injection of safety features into the parts of the design that are developing while simultaneously allowing for evaluation of the more mature parts of the design and the allocation of expectations for the next program milestone based on the results of those evaluations.

Refinements in selecting the actual safety review dates not only are required to optimize the available program PDR data being evaluated but also are used to maximize the time the program has to react to the half cycle milestone safety review results. The best way to achieve optimization of this approach is to first consider the needs of the safety process (note that in the first iteration of this new approach we based it on the needs of the program). The safety process needs as much next program milestone level data as possible while affecting this same milestone design. Starting the safety assessment exactly halfway between the milestones and allowing 60 days to cycle that process, would shift the actual safety review to be 90 – 120 days off the midpoint between the program milestones. However since a typical large scale program has at least one year between program major milestones this is readily accommodated by a program.

In particular this approach is of most benefit to very large scale integrations. Transfer of requirements, design and manufacturing data follows a long trail of asynchronous successive lower level project, element or system milestones and must be supported by the same design and hazard analysis personnel at the program milestones, moving the safety review processes milestones to these “between” program milestones creates flexibility in the overall program scheduling and allows for optimization of resources.

C. Risks Associated with the Half Cycle Safety Review Proposal

There are a few risks that need considered in modifying the timing of the currently standardized safety review process. First of all, there can be a perception that the safety reviews themselves are no longer part of the program milestones. It is up to the program safety, reliability and quality assurance (SR&QA) and program managers to assure that the successful completion of the half cycle milestone safety reviews be a part of the program milestone success criteria. In addition, to mitigate this perception, depending on program size and complexity, it may be wise to hold a mini phase review meeting that updates the status of each hazard cause from a relative risk standpoint for just a day or so during the program milestone period. This full phase risk update meeting would need very limited resources to support it and could then become the finalization of the half cycle review and closure for the program milestone review.

Meshing of the full phase criteria from the prior milestone data and the full phase criteria for the next milestone could be a complex challenge. If approached very literally, redefining each criterion for each hazard cause and program situation could result in meeting expectation conflicts between the safety review panel and the hardware developer or between supporting organizations. It is important here that the program and the SR&QA organizations meet and clearly define the data set maturity to be used for the safety review and that the reviewing panel understand the various levels of maturity in the design of that dataset before entering the review. This, however, is done by leveraging the program configuration management process to help define the relevant design data for the meeting. So the particular details of the design for each part of the program (project, element, system, subsystem) are predefined and understood. Knowing the program maturity in each area serves as the appropriate basis for determining the actual criteria for the half cycle milestone meeting. Clearly, all of the phase safety criteria for the prior program milestone applies but the SR&QA and hazard analysis teams will need to define what criteria from the subsequent program milestone is applicable to specific parts of the design. This whole process will have to iterate for each half cycle milestone safety review. However, as the program design matures and we reach the phase 3 closeout process, these criteria will naturally sync as there are no more forward milestone reviews after phase 3.

Finally, this process should end with a formal closeout that supports the program certification of flight readiness process or a Phase 4 safety review. This meeting would serve as a final closeout of hazard control verifications and allow for the safety panel and program design organizations to jointly buy off on the final safety products and design.

IV. Conclusion

Traditional safety review timing for large scale integrated systems has many mismatches that must be taken into account during the engineering and safety analysis processes. This paper presents a proposed approach to timing for formal safety reviews. The authors describe how the end products of the IHA development process are unviable using the traditional approach. This approach alleviates these problems and allows the hazard analysis process to occur as data normally matures.

References

[1]Enterprise Systems Optimization Laboratory, University of Illinois at Urbana-Champaign, source: https://netfiles.uiuc.edu/hmkim/www/research.html, 2009.

[2]CxP 70003, Constellation Program Plan, National Aeronautics and Space Administration, Washington, DC. July 2008.

[3]*Systems Engineering Fundamentals,* source: http://en.wikipedia.org/wiki/V-Model , Defense Acquisition University Press, 2001.

[4]*NASA Procedural Requirements 7120.5D*, National Aeronautics and Space Administration, Washington, DC, September 2009.

[5]Massie, M. J., “Constellation Integrated Hazard Analysis – Overcoming the Challenges,” Rome, Italy, October 2008.

[6]*Payload Safety Review and Data Submittal Requirements*, National Space Transportation System (NSTS) Payload Safety Review Process, NSTS 13830, National Aeronautics and Space Administration, Johnson Space Center, Houston, TX, July 1998.

[7]MIL-STD-882D, Standard Practice for System Safety, Department of Defense, Washington, DC., February 2000.

[8]Tethered Satellite, source: http://en.wikipedia.org/wiki/Tether_satellite, March, 2010.